



TUGAS AKHIR- KS 141501

PENGEMBANGAN APLIKASI BERGERAK PEMANDU PENGALAMAN KULTURAL DALAM KONTEKS BUDAYA JAWA

**RUSDI KURNIAWAN
NRP 5211 100 103**

**Dosen Pembimbing I
Arif Wibisono, S.Kom., M.Sc.**

**JURUSAN SISTEM INFORMASI
Fakultas Teknologi Informasi
Institut Teknologi Sepuluh Nopember
Surabaya 2016**



ITS
Institut
Teknologi
Sepuluh Nopember

FINAL PROJECT - KS 141501

MOBILE APPLICATION FOR CULTURAL EXPERIENCE GUIDE IN THE JAVANESE CONTEXT

RUSDI KURNIAWAN
NRP 5211 100 103

Academic Promotor I
Arif Wibisono, S.Kom., M.Sc.

INFORMATION SYSTEMS DEPARTMENT
Information Technology Faculty
Institut Teknologi Sepuluh Nopember
Surabaya 2016

KATA PENGANTAR

Puji syukur yang sebesar-besarnya Penulis panjatkan kehadirat Allah SWT, karena berkat rahmat dan hidayah-Nya Penulis dapat menyelesaikan Tugas Akhir yang merupakan salah satu syarat kelulusan pada Jurusan Sistem Informasi, Fakultas Teknologi Informasi, Institut Teknologi Sepuluh Nopember Surabaya.

Pengumpulan data Penulis dilapangan dilakukan selama kurun waktu 18 September 2015 hingga 20 Oktober 2015. Pengumpulan data dilakukan di Pasar Pabean, semua informasi yang ada di gali oleh penulis dengan cara mengamati langsung di lokasi yakni pasar Pabean surabaya. Mohon maaf apabila selama pengumpulan data ada beberoa pihak yang merasa terganggu atau waktunya tersita untuk membantu saya dalam proses pengerjaan Tugas Akhir

Atas berbagai bantuan, Penulis ingin menghaturkan rasa terimakasih yang sebesar-besarnya kepada:

- Bapak Aris Tjahyanto selaku Ketua Jurusan Sistem Informasi ITS selama saya manjalani kuliah di Sistem Informasi ITS. Terima kasih atas semua dukungan fasilitas selama ini.
- Bapak Arif Wibisono selaku pembimbing I dalam proses pengerjaan tugas akhir ini yang selalu bersedia waktunya diganggu untuk bimbingan tugas akhir. Terima kasih telah membimbing dengan sabar selama ini.
- Bapak Febrilian Samopa dan Bapak Nisfu Asrul Sani selaku penguji I dan penguji II yang bersedia menguji tugas akhir saya. Terima kasih meskipun berstatus penguji tetapi saya merasa seperti dibimbing.

- Bapak Radityo Prasetyanyo selaku dosen wali selama 3 tahun menjalani kuliah di jurusan Sistem Informasi ITS.
- Ibu Renny Pradina selaku dosen wali (peralihan) selama 2 semester menjalani kuliah di jurusan Sistem Informasi ITS.
- Ibu Nur Aini selaku penguji I yang bersedia menguji tugas akhir dalam sidang proposal dan sidang progress.
- Bapak Andre Parvian selaku penguji II yang bersedia menguji tugas akhir dalam sidang proposal dan sidang progress.
- Mas Bambang Wijanarko yang telah meluangkan waktu untuk sharing, dan mengatur waktu sidang di laboratorium E-Bisnis.
- Pihak pasar pabean yang mengizinkan proses pengambilan data sebagai bahan penelitian tugas akhir.

Penulis menyadari bahwa tugas akhir ini masih banyak kekurangan dan ketidaksempurnaan, untuk itu penulis mengharapkan saran atas tugas akhir ini yang bersifat membangun guna perbaikan di masa mendatang. Akhir kata, penulis berharap tugas akhir ini dapat bermanfaat bagi kita semua.

Surabaya, 08 Januari 2016

Penulis

LEMBAR PENGESAHAN

PENGEMBANGAN APLIKASI BERGERAK PEMANDU PENGALAMAN KULTURAL DALAM KONTEKS BUDAYA JAWA

TUGAS AKHIR

Disusun Untuk Memenuhi Salah Satu Syarat
Memperoleh Gelar Sarjana Komputer
pada

Jurusan Sistem Informasi
Fakultas Teknologi Informasi
Institut Teknologi Sepuluh Nopember

Oleh :

RUSDI KURNIAWAN
NRP 5211 100 103

Surabaya, 08 Januari 2016

KETUA
JURUSAN SISTEM INFORMASI



Ir. Aris Tjahyanto, M.kom.

NIP 19650310 199102 1 001

LEMBAR PERSETUJUAN

PENGEMBANGAN APLIKASI BERGERAK PEMANDU PENGALAMAN KULTURAL DALAM KONTEKS BUDAYA JAWA

TUGAS AKHIR

**Disusun Untuk Memenuhi Salah Satu Syarat
Memperoleh Gelar Sarjana Komputer
pada
Jurusan Sistem Informasi
Fakultas Teknologi Informasi
Institut Teknologi Sepuluh Nopember**

Oleh :


RUSDI KURNIAWAN

NRP 5211 100 103

Disetujui Tim Penguji : Tanggal Ujian : 08 Januari 2016

Periode Wisuda : Maret 2016

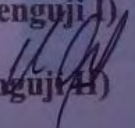
Arif Wibisono, S.Kom., M.Sc.


(Pembimbing I)

Dr.Eng. Febriliyan Samopa, S.Kom., M.Kom.


(Penguji I)

Nisfu Asrul Sani, S.Kom., M.Sc.


(Penguji II)

PENGEMBANGAN APLIKASI BERGERAK PEMANDU PENGALAMAN KULTURAL DALAM KONTEKS BUDAYA JAWA

Nama Mahasiswa : Rusdi Kurniawan
NRP : 5211 100 103
Jurusan : Sistem Informasi FTIF ITS
Dosen Pembimbing I: Arif Wibisono, S.Kom.,M.Sc.

Abstrak

Industri pariwisata adalah industri yang memiliki potensi besar dalam meningkatkan pendapatan sebuah negara dan mengurangi angka pengangguran, Salah satu sektor yang memiliki kontribusi besar terhadap industry pariwisata adalah pariwisata budaya. Di sini, Uni Eropa mencatat bahwa 40 persen pariwisata dunia adalah pariwisata dari sisi budaya. Melihat kondisi tersebut banyak potensi yang bisa digali untuk mengoptimalkan pariwisata daerah. Indonesia adalah salah satu negara dengan potensi besar sebagai penyedia pariwisata budaya karena keragaman suku dan budaya.

Salah satu jenis wisatawan budaya yang memiliki potensi untuk dikembangkan adalah purposeful cultural tourist. Wisatawan jenis ini adalah wisatawan yang menikmati budaya dari sisi intelektual. Sekalipun wisatawan jenis ini memiliki nilai potensial yang besar, namun belum ada aplikasi yang membantu secara khusus bagi wisatawan dengan tipe ini.

Tugas akhir ini bertujuan untuk membuat purwarupa aplikasi yang dapat membantu meningkatkan pengalaman belajar budaya bagi wisatawan dengan tipe purposeful cultural tourist. Sehingga para wisatawan tersebut bisa mendapatkan pengalaman belajar budaya dengan mudah dan nyaman. Dengan begitu diharapkan ketika wisatawan akan datang ke suatu daerah sudah mempunyai pandangan mengenai budaya yang ada di daerah tersebut.

Aplikasi ini berbasis mobile dengan suasana pasar Pabean Surabaya dan menggunakan bahasa jawa surabaya, dengan informasi yang diberikan mengenai budaya jawa surabaya. Pemilihan bahasa jawa sendiri juga dipilih karena kemudahan akses masyarakat dan infrastruktur teknologi informasi. Pasar dipilih karena keragaman dan interaksi kultur yang bisa dieksplorasi. Tugas akhir ini bermanfaat bagi para purposeful cultural tourist sebagai pemandu pengalaman budaya dan pemerintah dalam peningkatan jumlah wisatawan.

Aplikasi berbasis mobile karena untuk memudahkan aksesibilitas, selain itu dengan mobile akan lebih flexible dibandingkan berbasis web atau desktop. Web mungkin bisa dikatakan mudah untuk tahap pengembangan ke tahap selanjutnya, namun disini juga ada pertimbangan masalah aksesibility, terutama di daerah yang sulit jaringan.

Kata Kunci: Budaya Jawa, Aplikasi Bergerak, Pasar, Purposeful Cultural Tourist, Carpooling, Google maps.

MOBILE APPLICATION FOR CULTURAL EXPERIENCE GUIDE IN THE JAVANESE CONTEXT

Nama Mahasiswa : Rusdi Kurniawan
NRP : 5211 100 103
Jurusan : Sistem Informasi FTIF ITS
Dosen Pembimbing I: Arif Wibisono, S.Kom.,M.Sc.

Abstract

Tourism is a potential industry in raising income for a country and decreasing the number of unemployed people. One of sectors which has big contribution for tourism industry is cultural tourism. Here, European Union noted that 40% of tourism of the world mostly comes from culture. By seeing this situation, there will be much potential we get to develop regional tourism. Indonesia is a country with big potential of cultural potential from its various races and cultures.

One of cultural tourism which has potential to develop is *purposeful cultural tourist*. This kind of tourists prefer to enjoy culture from its intellectual side. Although these tourists really have big potential value, however there is still no supporting applications to help them.

This study aims to make PURWARUPA of application which is able to raise cultural learning experience for *purposeful cultural tourist*. So, those tourists will achieve more comfortable and easier cultural learning experience. Then, we hope all of the tourists who will visit some places have known all about the culture there.

This application is in a form of mobile apps with background of a situation in Pabean market of Surabaya, Javanese language, and also completed by information about Surabaya Javanese culture. We choose Javanese language because of its easy accessibility and information technology infrastructure. Market situation is chosen because of its cultural variety and interaction which can be explored. This study is necessary for *purposeful cultural tourist* as a guidance of cultural experience and government for increasing the number of tourists' coming.

This mobile apps is created to make the accessibility easier, while it will be more flexible than web or desktop. Web is possibly easy for the next development, yet here we also consider about the accessibility problem, especially for some remote areas.

Keywords: Budaya Jawa, Aplikasi Bergerak, Pasar, Purposeful Cultural Tourist, Carpooling, Google maps

DAFTAR ISI

TUGAS AKHIR.....	ii
Abstrak	i
Abstract	iii
KATA PENGANTAR	v
DAFTAR ISI.....	vii
DAFTAR GAMBAR	xv
DAFTAR TABEL.....	xix
BAB I PENDAHULUAN	1
1.1 Latar Belakang Masalah	1
1.2 Rumusan Masalah.....	8
1.3 Batasan Masalah	8
1.4 Tujuan	8
1.5 Manfaat	9
1.6 Relevansi.....	9
BAB II TINJAUAN PUSTAKA.....	11
2.1. Studi Sebelumnya	11
2.2. Pariwisata Budaya.....	12
2.3. Pasar Tradisional.....	14
2.4. Pasar Pabean Surabaya	15
2.5. Wisata Surabaya.....	17

2.6. Android	19
2.6.1. Android SDK (Software Development Kit)	20
2.6.2. ADT (Android Development Tools)	21
2.6.3. Eclipse	22
2.7. Iterative Waterfall Model	22
2.8. XML	23
2.8.1. XML untuk Android	24
BAB III METODOLOGI	25
3.1. Tahapan Pelaksanaan Tugas Akhir	26
3.1.1. Studi Literatur	26
3.1.2. Identifikasi Kebutuhan Perangkat Lunak	26
3.1.3. Perancangan Aplikasi	26
3.1.4. Pengkodean Aplikasi	28
3.1.5. Pengujian Aplikasi	29
3.1.6. Penyusunan Laporan Tugas Akhir	29
3.1.7. Jadwal Pelaksanaan	29
BAB IV PERANCANGAN	31
4.1. Kebutuhan Fungsional	31
4.2. Aktor-Aktor	32
4.3. Use Case	32
4.4. Activity Diagram	33
4.5. Sequence Diagram	34
4.6. Class Diagram	34
4.7. Entity Relationship Diagram (ERD)	35

4.8. Penggunaan Test Case	35
BAB V IMPLEMENTASI	37
5.1. Lingkungan Implementasi	37
5.2. Struktur Direktori	38
5.3. Implementasi Aplikasi Cultural Experience Guide...	38
5.3.1. Antar Muka Halaman Utama	38
5.3.2. Kode Progam Halaman Utama	41
5.3.3. Mencari Informasi Produk	45
5.3.4. Melihat Titik Lokasi Pedagang	51
5.3.5. Memberikan Informasi Etika di Pasar	52
5.3.6. Memberikan Informasi Dialek Jawa di Pasar	53
5.3.7. Memberikan Informasi Dialek Pasar Antara Pedagang dan Pembeli	54
5.3.8. Melihat dan Menentukan Informasi Petunjuk Arah ke Pasar Pabean Surabaya.....	56
5.4. Uji Coba	58
5.4.1. Uji Coba Fungsional	58
5.4.2. Uji Coba Performa	59
5.4.3. Uji Coba Penerimaan pengguna.....	61
BAB VI HASIL DAN PEMBAHASAN	63
6.1. Uji Coba	63
6.1.1. Uji Coba Fungsional	63
6.1.2. Hasil Uji Coba Performa.....	64
6.1.3. Hasil Uji Coba Penerimaan Pengguna	67
6.2. Pembahasan Uji Coba	69

6.2.1. Pembahasan Hasil Uji Coba Fungsional	69
6.2.2. Pembahasan Hasil Uji Coba Performa	70
6.2.3. Pembahasan Hasil Uji Coba Penerimaan Pengguna	70
BAB VII KESIMPULAN DAN SARAN	73
7.1. Kesimpulan	73
7.2. Saran.....	74
Daftar Pustaka	75
BIODATA PENULIS.....	77
LAMPIRAN A	A1
A.1. Use Case Model untuk Wisatawan.....	A1
LAMPIRAN B	B1
AC01-1-01 Mencari informasi produk	B1
AC01-1-02 Mencari informasi penjual	B2
AC02-1-01 Melihat informasi penting terkait kriminalitas	B3
AC03-1-01 Melihat titik lokasi pedagang di denah.....	B4
AC03-1-02 Melihat dan menentukan informasi petunjuk arah ke Pasar Pabean	B5
AC04-1-01 Memberikan informasi etika saat di pasar.....	B6
AC05-1-01 Memberikan informasi Dialek Jawa di Pasar..	B7
AC05-1-02 Memberikan informasi Dialek Pasar yang dilakukan pedagang dan pembeli	B8
AC05-1-03 Melihat informasi ekspresi-ekspresi dialek angka Jawa.	B9
AC06-1-01 Mencari dan melihat informasi harga terkini di Pasar Pabean Surabaya	B10

AC07-1-01 Melihat informasi suku dan kebiasaan pedagang di Pasar Pabean Surabaya.....	B11
LAMPIRAN C	C1
SD01-1-01 Mencari informasi produk	C1
SD01-1-02 Mencari informasi penjual.....	C2
SD02-1-01 Melihat informasi penting terkait kriminalitas	C3
SD03-1-01 Melihat titik lokasi pedagang di denah.....	C4
SD03-1-02 Melihat dan menentukan informasi petunjuk arah ke Pasar Pabean.....	C5
SD04-1-01 Memberikan informasi etika saat di pasar	C6
SD05-1-01 Memberikan informasi Dialek Jawa di Pasar ..	C7
SD05-1-02 Memberikan informasi Dialek Pasar yang dilakukan pedagang dan pembeli	C8
SD05-1-02 Memberikan informasi Dialek Pasar yang dilakukan pedagang dan pembeli (lanjutan).....	C9
SD05-1-02 Memberikan informasi Dialek Pasar yang dilakukan pedagang dan pembeli (lanjutan).....	C10
SD05-1-02 Memberikan informasi Dialek Pasar yang dilakukan pedagang dan pembeli (lanjutan).....	C11
LAMPIRAN D	D1
Class Diagram Aplikasi <i>Cultural Experience Guide</i>	D1
LAMPIRAN E	E1
Entity Relationship Diagram Aplikasi <i>Cultural Experience Guide</i>	E1
LAMPIRAN F	F1
<i>User Acceptance Test (UAT)</i>	F1

LAMPIRAN G	G1
Test Case	G1
LAMPIRAN H <i>SOURCE CODE</i>	H1
1.) Layout splashscreen.xml	H1
2.) Potongan kode <i>class</i> rinci.....	H2
3.) Potongan kode <i>layout</i> main	H6
4.) Potongan kode <i>class</i> Main Activity	H7
5.) Potongan kode <i>class</i> rinci.....	H8
6.) Potongan kode <i>class</i> Splashscreen.	H10
7.) Potongan kode <i>class</i> rinci menampilkan informasi produk.....	H13
8.) Potongan kode <i>class</i> Arti menampilkan informasi produk.....	H15
9.) Potongan kode <i>layout</i> arti.....	H23
10.) Potongan kode <i>class</i> arti untuk menampilkan detail produk.....	H26
11.) Potongan kode <i>class</i> MainActivity untuk menampilkan tab menu pedagang.	H27
12.) Potongan kode <i>class</i> cari.java pencarian pedagang/penjual.....	H29
13.) Potongan kode <i>detail</i> pedagang/penjual.....	H30
14.) Potongan kode <i>class</i> listpeta.	H32
15.) Potongan kode informasi pedagang (Salah satu contoh pedagang Ikan Tutik IK01A).....	H34
16.) Potongan kode <i>class</i> EtikaPasar	H40
17.) Potongan kode <i>class</i> KamusDialek.	H46

- 18.) Potongan kode menampilkan informasi dialek pasar antara pedagang dan pembeli, contohnya salah satu contohnya pedagang ikan yakni *class* DialekIkan. H52
- 19.) Potongan kode menampilkan informasi dialek pasar antara pedagang dan pembeli, contohnya salah satu contohnya pedagang buah yakni *class* DialekBuah. H78
- 20.) Potongan kode *class* PsPabean. H104
- 21.) Potongan kode RutePerjalanan.html H104

(halaman ini sengaja dikosongkan)

DAFTAR GAMBAR

Gambar 1. 1 The typology of cultural tourist by McKercher and Du Cros.....	5
Gambar 1. 2 Statistik Frekuensi Penggunaan Smartphone (sumber: Flurry Analytics).....	7
Gambar 1. 3 Pohon Penelitian Laboratorium E-Business.....	10
Gambar 3. 1 Tahapan Pengerjaan Tugas Akhir	25
Gambar 3. 2 Use Case Diagram	27
Gambar 3. 3 Activity Diagram User	27
Gambar 3. 4 Arsitektur Sistem.....	28
Gambar 3. 5 Jadwal Pelaksanaan Pengerjaan Tugas Akhir. ...	30
Gambar 5. 1 Struktur Direktori Project.....	38
Gambar 5. 2 Tampilan layout splashscreen	39
Gambar 5. 3 Tampilan <i>layout</i> rinci	40
Gambar 5. 4 Tampilan <i>layout</i> main.....	41
Gambar 5. 5 Potongan kode <i>class</i> rinci.....	42
Gambar 5. 6 <i>Class</i> rinci dan <i>method</i> memanggil <i>class</i> lain. ...	43
Gambar 5. 7 Method menentukan database.	44
Gambar 5. 8 Potongan kode method <i>InitDatabase Class</i> Splashscreen.....	44
Gambar 5. 9 Method <i>Action button</i> klik daftar produk	45
Gambar 5. 10 Potongan kode rinci.java memanggil <i>class</i> Arti.java.....	45
Gambar 5. 11 Potongan kode Arti.java menampilkan data yang ada di database.....	46
Gambar 5. 12 Tampilan <i>layout</i> arti.xml	47
Gambar 5. 13 Potongan kode pencarian produk	48
Gambar 5. 14 Potongan kode halaman detail produk	49
Gambar 5. 15 Potongan kode tab menu pedagang atau produk	49

Gambar 5. 16 Potongan kode pencarian pedagang atau produk.	50
Gambar 5. 17 Potongan kode detail pedagang atau produk ...	51
Gambar 5. 18 Potongan kode menampilkan daftar peta.	51
Gambar 5. 19 Potongan kode menampilkan isi peta yang dipilih (Contoh pedagang Ikan).	52
Gambar 5. 20 Potongan kode EtikaPasar.java.	52
Gambar 5. 21 Method untuk megambil data di <i>class</i> KamusDialek	53
Gambar 5. 22 Potongan kode menampilkan kumpulan Dialek Jawa.	54
Gambar 5. 23 Potongan kode menampilkan dialek Jawa Pedagang Ikan.	55
Gambar 5. 24 Potongan kode menampilkan dialek Jawa pedagang buah.	55
Gambar 5. 25 Potongan kode menampilkan dialek Jawa pedagang sayur	56
Gambar 5. 26 Potongan kode memanggil halaman HTML.	57
Gambar 5. 27 Potongan kode halaman HTML untuk menggunakan Goole Maps API.	57
Gambar 5. 28 Potongan kode progam untuk petunjuk arah ke Pasar Pabean Surabaya	58
Gambar 5. 29 Aplikasi Uji Tes Benchmark AnTutu Benchmark	60
Gambar A. 1 Use Case Model Wisatawan	A1
Gambar B. 1 Activity Diagram mencari informasi produk ..	B1
Gambar B. 2 Activity Diagram mencari Informasi penjual ..	B2
Gambar B. 3 Activity Diagram melihat informasi penting terkait kriminalitas.	B3
Gambar B. 4 Activity Diagram melihat titik lokasi pedagang di denah	B4

Gambar B. 5 Activity Diagram melihat dan menentukan informasi petunjuk arah ke Pasar Pabean.....	B5
Gambar B. 6 Activity Diagram memberikan informasi etika saat di pasar	B6
Gambar B. 7 Activity Diagram memberikan informasi Dialek Jawa di Pasar	B7
Gambar B. 8 Activity Diagram memberikan informasi Dialek Pasar jual-beli.....	B8
Gambar B. 9 Melihat informasi ekspresi-ekspresi dialek angka Jawa	B9
Gambar B. 10 Mencari dan melihat informasi harga terkini di Pasar Pabean Surabaya.....	B10
Gambar B. 11 Melihat informasi suku dan kebiasaan pedagang di Pasar Pabean Surabaya.....	B11
Gambar C. 1 Sequence Diagram mencari informasi produk. C1	
Gambar C. 2 Sequence Diagram Mencari informasi penjual. C2	
Gambar C. 3 Sequence Diagram melihat informasi penting terkait kriminalitas	C3
Gambar C. 4 Sequence Diagram melihat lokasi pedagang di denah	C4
Gambar C. 5 Sequence Diagram melihat dan menentukan informasi petunjuk arah ke Pasar Pabean.....	C5
Gambar C. 6 Sequence Diagram memberikan informasi etika saat di pasar	C6
Gambar C. 7 Sequence Diagram memberikan informasi dialek Jawa di Pasar	C7
Gambar D. 1 Class Diagram Aplikasi	D1
Gambar E. 1 ERD aplikasi <i>Cultural Experience Guide</i>	E1

(halaman ini sengaja dikosongkan)

DAFTAR TABEL

Tabel 1.1 Perkembangan Wisatawan Mancanegara Ke Indonesia	3
Tabel 1. 2 Jenis Tourist menurut McKercher Dan Du Cros.....	5
Tabel 4. 1 Daftar kebutuhan fungsional	31
Tabel 4. 2 Daftar Aktor	32
Tabel 4. 3 Daftar Use Case.....	32
Tabel 4. 4 Daftar Aktivitas	33
Tabel 5. 1 Spesifikasi Perangkat Keras	37
Tabel 5. 2 Spesifikasi Perangkat Lunak	37
Tabel 5. 3 Format Test Case.....	58
Tabel 5. 4 Daftar Test Case.....	59
Tabel 6. 1 Hasil Uji Coba Fungsional	63
Tabel 6. 2 Hasil Uji Coba Tiap Fitur.....	67
Tabel 6. 3 Hasil Uji Coba untuk Responden	68
Tabel 6. 4 Hasil <i>feedback</i> pengguna.....	68
Tabel G. 1 Test Case Load Halaman Utama	G1
Tabel G. 2 Test Case Melihat Daftar Produk	G1
Tabel G. 3 Test Case Melihat Daftar Pedagang	G2
Tabel G. 4 Test Case Mencari Nama Produk	G2
Tabel G. 5 Test Case Mencari Nama Pedagang	G3
Tabel G. 6 Test Case Melihat dan Mendengarkan Kumpulan Dialek Jawa	G3
Tabel G. 7 Test Case Melihat dan Mendengarkan Dialek Jawa Pedagang Ikan.	G4
Tabel G. 8 Test Case Melihat dan Mendengarkan Dialek Jawa Pedagang Buah.	G5
Tabel G. 9 Test Case Melihat dan Mendengarkan Dialek Jawa Pedagang Sayur.....	G6

Tabel G. 10 Test Case Melihat Peta Lokasi dan Petunjuk Arah ke Pasar Pabean	G7
Tabel G. 11 Test Case Melihat Denah Pasar Pabean Surabaya.	G8
Tabel G. 12 Melihat lokasi pedagang	G8
Tabel G. 13 Melihat Informasi Etika di Pasar.	G9
Tabel G. 14 Test Case Melihat Informasi Terkait Kriminalitas di Pasar Pabean Surabaya.	G9
Tabel G. 15 Melihat Informasi Suku dan Kebiasaan di Pasar Pabean Surabaya.....	G10
Tabel G. 16 Test Case Melihat dan Mencari Harga Komoditi Produk di Pasar Pabean Surabaya.	G11
Tabel G. 17 Test Case Melihat dan Mendengarkan Angka Jawa Surabaya.	G12

BAB I

PENDAHULUAN

Pada bab ini, akan dijelaskan tentang Latar Belakang Masalah, Perumusan Masalah, Batasan Masalah, Tujuan Tugas Akhir, Relevansi dan Manfaat Kegiatan Tugas Akhir.

1.1 Latar Belakang Masalah

Pariwisata dianggap sebagai salah satu sektor ekonomi paling penting bagi suatu negara. Disamping sebagai mesin penggerak ekonomi, pariwisata juga merupakan wahana yang menarik untuk mengurangi angka pengangguran [1]. Dalam hal ini, peran pemerintah sebagai fasilitator perlu menjamin terlaksananya pembangunan. Dengan adanya pembangunan dan perkembangan industri pariwisata di suatu wilayah, arus urbanisasi ke kota-kota besar dapat lebih ditekan. Hal ini disebabkan karena pariwisata memiliki tiga aspek pengaruh yaitu aspek ekonomis sebagai sumber devisa dan pajak-pajak, aspek sosial sebagai penciptaan lapangan kerja dan aspek budaya. Pariwisata juga merupakan suatu sektor yang tidak jauh berbeda dengan sektor ekonomi yang lain yaitu dalam proses perkembangannya juga mempunyai dampak atau pengaruh dibidang sosial dan ekonomi. [2]. Uni Eropa mencatat bahwa 40 persen pariwisata dunia adalah pariwisata dari sisi budaya. Di sini, kami melihat bahwa Indonesia adalah salah satu negara dengan potensi besar sebagai penyedia pariwisata budaya karena keragaman suku dan budaya [3].

Indonesia merupakan salah satu negara yang memiliki sejarah dan kebudayaan yang sangat beranekaragam [4]. Berbagai atraksi budaya ini dapat dinikmati wisatawan nusantara dan mancanegara sebagai salah satu wisata heritage (warisan budaya) yang berada di Indonesia. Dengan adanya kunjungan wisatawan ini, akan menambah pendapatan daerah yang dikunjungi. Terlebih bagi wisatawan mancanegara yang

datang, tentu akan mendatangkan devisa bagi negara. Hal ini tentunya akan menjadi acuan bagi daerah-daerah untuk dapat mengembangkan dan mengelola pariwisata agar dapat mendatangkan sumber dana untuk sektor pendapatan asli daerah. Industri pariwisata merupakan jenis industri yang mempunyai rantai kegiatan yang sangat panjang. Banyak kegiatan yang terkait dengan industri pariwisata, seperti kegiatan biro perjalanan wisata, transportasi, perhotelan, restoran, kesenian dan budaya daerah, kerajinan rakyat, pemandu untuk mamandu wisatawan mancanegara, pameran dan olahraga internasional yang diselenggarakan di daerah-daerah, dan kegiatan-kegiatan lainnya.

Dari seluruh jumlah angka kunjungan wisatawan mancanegara dan nusantara, hampir separuhnya mengunjungi objek wisata yang terkait dengan seni budaya. Jumlah kunjungan wisatawan mancanegara (wisman) ke Indonesia pada Januari 2015 mencapai 723,0 ribu kunjungan atau turun 3,99 persen dibandingkan jumlah kunjungan wisman Januari 2014, yang tercatat sebanyak 753,1 ribu kunjungan. Demikian pula jika dibandingkan dengan Desember 2014, jumlah kunjungan wisman Januari 2015 turun sebesar 21,01 persen [5]. Daerah Bali dan Yogyakarta sekitarnya merupakan obyek wisata budaya favorit yang dijadikan sasaran kunjungan para wisatawan. Jika dilihat dari segi potensi pemanfaatan, sebenarnya masih terdapat banyak daerah yang bisa diangkat dan dikembangkan sebagai lokasi wisata budaya. Untuk data terbaru dapat dilihat di tabel 1.1 mengenai pertumbuhan kunjungan wisatawan mancanegara ke Indonesia.

Tabel 1. 1Perkembangan Wisatawan Mancanegara Ke Indonesia

Jumlah Kunjungan Wisman Melalui 19 Pintu Masuk Utama		Pertumbuhan (%)
2015	2014	
105,158	117,207	-10.28
83,834	105,410	-20.94
37,092	34,520	7.45
37,089	33,106	12.03
13,913	16,851	-17.44
80,842	88,583	-8.74
17,499	17,905	-2.27
8,834	9,932	-11.06
5,231	6,848	-23.61
5,979	5,777	3.50
90,935	83,700	8.64
16,702	17,615	-5.18
16,036	15,397	4.15
10,621	10,733	-1.04
9,129	10,121	-9.80

Jumlah Kunjungan Wisman Melalui 19 Pintu Masuk Utama		Pertumbuhan (%)
2015	2014	
10,466	10,683	-2.03
6,858	12,940	-47.00
680	583	16.64
474	527	-10.06
104	154	-32.47
102,407	102,697	-0.28
671,939	716,419	-6.21
51,100	36,660	39.39
723,039	753,079	-3.99

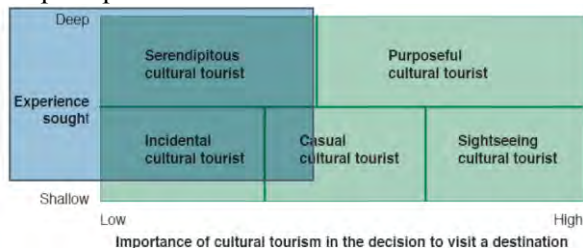
Sumber : Ditjen Imigrasi dan BPS (diolah kembali oleh Pusdatin Kemenparekraf)

Catatan : Data kebangsaan tahun 2015 merupakan data sementara, karena termasuk data kebangsaan dari pintu lainnya.

Keterangan jenis pintu masuk : U (Udara), L (Laut), D (Darat)

Wisata budaya mungkin didefinisikan sebagai gerakan yang melibatkan orang dalam eksplorasi atau pengalaman yang beragam cara hidup orang lain, mencerminkan semua kebiasaan sosial, tradisi, keagamaan, atau ide-ide intelektual warisan budaya mereka –(ICOMOS Charter for Cultural Tourism, Draft April 1997) [6]. Menurut McKercher

dan DuGros yang membedakan lima jenis wisatawan budaya berdasarkan pentingnya budaya keputusan mereka untuk melakukan perjalanan dan juga kedalaman pengalaman mereka seperti pada Gambar 1.1 .



Gambar 1. 1 The typology of cultural tourist by McKercher and Du Cros

Untuk definisinya mengenai lima jenis wisatawan budaya menurut Mckercher dan Du Gros dapat dilihat pada tabel 1.2.

Tabel 1. 2 Jenis Tourist menurut McKercher Dan Du Cros

Type of cultural tourist	Short characterisation
The purposeful cultural tourist	Cultural tourism is the primary motivation for visiting a destination and the tourist has a very deep and elaborate cultural experience
The sightseeing cultural tourist	Cultural tourism is a primary reason for visiting a destination, but the experience is less deep and elaborated
The serendipitous cultural tourist	A tourist who does not travel for cultural reasons, but who, after participating, ends up having a deep cultural tourism experience
The casual cultural tourist	Cultural tourism is a weak motive for travel and the resulting experience is shallow
The incidental cultural tourist	This tourist does not travel for cultural reasons, but nonetheless participates in some activities and has shallow experiences

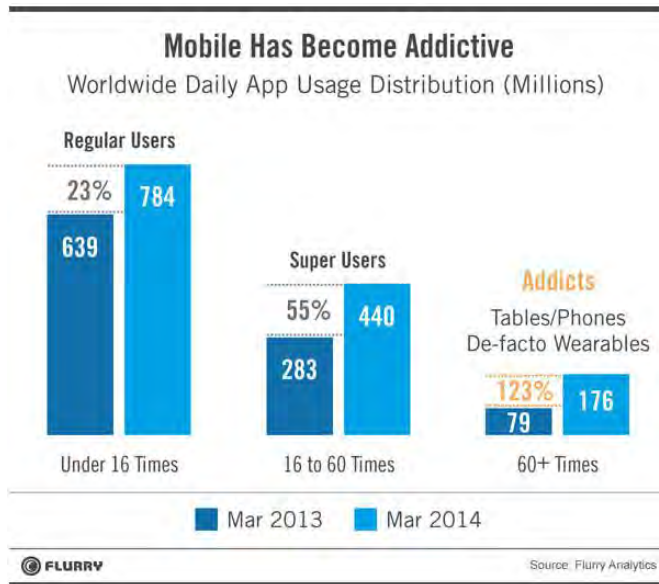
Berdasarkan pembagian lima jenis wisatawan budaya pada Tabel 1.2, salah satu jenis wisatawan budaya yang memiliki potensi untuk dikembangkan adalah purposeful cultural tourist. Wisatawan ini merupakan wisatawan yang memiliki motivasi besar terhadap kunjungan wisata dan memiliki keingintahuan yang dalam mengenai pengalaman budaya Walaupun Tidak ada angka pasti yang mengukur jumlah

wisatawan dengan kategori ini, akan tetapi wisatawan jenis purposeful cultural tourist memiliki potensi untuk dikembangkan. Namun belum ada aplikasi yang membantu secara khusus bagi wisatawan dengan tipe ini. Diharapkan nantinya dengan tugas akhir ada aplikasi yang dapat membantu meningkatkan pengalaman belajar budaya bagi wisatawan dengan tipe purposeful cultural tourist. Sehingga para wisatawan tersebut bisa mendapatkan pengalaman belajar budaya dengan mudah dan nyaman.

Dalam studi kasus ini menggunakan segment pasar dan menggunakan budaya Jawa. Pemilihan pasar sendiri karena banyak transaksi yang dilakukan dengan karakteristik budaya yang berbeda-beda oleh setiap orangnya. Pasar tidak semata mawadahi kegiatan ekonomi, akan tetapi pelaku juga dapat menjadikan wisata budaya. Dengan demikian dapat dikatakan bahwa pasar tradisional dapat menjadi wadah kegiatan ekonomi, interaksi sosial, dan sarana rekreasi baik suasana pasar maupun produk barang dagangan yang khas [7]. Untuk budaya Jawa dipilih karena kemudahan akses masyarakat dan infrastruktur teknologi informasi. Penggunaan teknologi dapat membantu pengelolaan informasi yang dibutuhkan dengan cepat, tepat, dan akurat.

Penggunaan smartphone semakin meningkat dari tahun ke tahun, selain sebagai media komunikasi, smartphone memiliki fitur yang bisa dinikmati semua strata social masyarakat baik yang kaya maupun kalangan dari ekonomi menengah ke bawah. Dengan begitu segala keperluan kita sehari-hari hampir semuanya dilakukan melalui perangkat mobile. Berdasarkan kondisi ini hal ini perusahaan periklanan mobile Flurry mengeluarkan laporan mengenai jumlah pecandu smartphone saat ini. Dalam laporannya, Flurry mematok bahwa Mobile Addict atau pecandu smartphone adalah orang yang membuka aplikasi pada smartphone mereka sebanyak lebih dari 60 kali dalam sehari. Hasilnya cukup menarik. Dari

1,4 miliar pengguna smartphone yang diteliti, 176 juta orang di antaranya adalah pecandu smartphone. Angka tersebut juga naik sampai 123 persen dibandingkan angka tahun lalu yang hanya 79 juta orang [8].



Gambar 1. 2 Statistik Frekuensi Penggunaan Smartphone
(sumber: Flurry Analytics)

Tinjauan terhadap perkembangan teknologi inilah yang melahirkan suatu gagasan ide untuk menciptakan suatu aplikasi mobile yang dapat memberikan warga kota dan wisatawan untuk meningkatkan pengalaman belajar budaya. Mengingat masyarakat sekarang yang sudah terintegrasi dengan teknologi dalam beberapa aspek, khususnya dalam mencari informasi dan juga penggunaan perangkat mobile yang semakin banyak digunakan, sehingga pengenalan informasi berupa digital dibutuhkan di era ini.

Aplikasi ini nantinya bisa digunakan oleh semua orang yang ingin berkunjung suatu daerah, khususnya untuk para wisatawan. Aplikasi ini bertujuan memberikan pengalaman budaya kepada wisatawan dengan tipe *purposeful cultural tourist* ketika berkunjung ke suatu daerah. Dengan aplikasi ini wisatawan akan dengan mudah mempelajari budaya yang ada, sehingga ketika berkunjung wisatawan dapat menambah pengalaman kultural dan menerapkannya di daerah tersebut.

1.2 Rumusan Masalah

Rumusan masalah adalah bagaimana mengembangkan purwarupa pemandu pengalaman kultural berbasis mobile.

1.3 Batasan Masalah

Batasan masalah pada tugas akhir ini adalah:

- 1.) Hanya ada satu pasar tradisional sebagai model yakni Pasar Pabean Surabaya.
- 2.) Penggunaan bahasa di aplikasi menggunakan bahasa Jawa dengan dialek Surabaya.
- 3.) Aplikasi pemandu menggunakan bahasa pengantar Bahasa Indonesia.
- 4.) Diasumsikan tidak ada perubahan lokasi pedagang di pasar
- 5.) Diasumsikan sinyal internet berjalan lancar.
- 6.) Tidak mencakup shortest path

1.4 Tujuan

Adapun tujuan dari tugas akhir ini, diantaranya adalah:

- 1.) Membuat purwarupa aplikasi pemandu pengalaman kultural dengan konteks budaya Jawa
- 2.) Menggali informasi potensi sejarah dan budaya pasar tradisional sebagai tujuan wisata.

1.5 Manfaat

Manfaat dan kegunaan dari tugas akhir ini adalah :

A.) Manfaat kepada pemerintah

Manfaat bagi pemerintah antara lain:

1. Aplikasi ini membantu meningkatkan potensi pariwisata di daerah
2. Aplikasi ini meningkatkan kualitas sistem dan pelayanan sektor pariwisata khususnya wisata budaya.
3. Aplikasi ini meningkatkan kondisi ekonomi daerah, serta meningkatkan angka kunjungan wisatawan terhadap wisata budaya yang dimiliki daerah tersebut.

B.) Manfaat kepada wisatawan

Manfaat bagi wisatawan antara lain:

1. Meningkatkan pengalaman belajar budaya yang menantang secara intellectual khususnya wisatawan jenis *purposeful cultural tourist*.
2. Memudahkan wisatawan mendapatkan informasi terkait pengalaman budaya yang ada di daerah yang sedang dikunjunginya

1.6 Relevansi

Relevansi penelitian tugas akhir ini terhadap penelitian area sistem informasi berada pada area perencanaan proses *e-business* dengan topik perencanaan *e-business* sebagaimana digambarkan pada Gambar 1.3. Adapun beberapa mata kuliah yang terkait dengan penelitian ini adalah Algoritma Pemrograman 2, Manajemen Basis Data, Rekayasa Kebutuhan Perangkat Lunak, Rancang Bangun Perangkat Lunak dan Tata Tulis Ilmiah.

Pohon Penelitian Laboratorium E-Bisnis



Gambar 1. 3 Pohon Penelitian Laboratorium E-Business.

BAB II

TINJAUAN PUSTAKA

Pada bagian ini akan dibahas mengenai hasil studi penelitian sebelumnya dan teori-teori yang mendukung dalam pengerjaan tugas akhir.

2.1. Studi Sebelumnya

Untuk penelitian secara khusus mengenai pemandu wisata budaya terhadap *purposeful cultural tourist* belum ada. Namun Ada beberapa penelitian terkait dengan panduan wisata secara online. Salah satunya penelitian mengenai panduan lokasi objek wisata dengan google Maps dan GPS untuk Wisata Kota Solo yang pernah dilakukan oleh Jati K., Ringga [9]. Perkembangan multimedia membuat penyampaian informasi menjadi lebih praktis. Solo merupakan sebuah kota yang memiliki potensi wisata yang merupakan peninggalan sejarah. Masalahnya adalah media untuk mempromosikan aset wisata tersebut belum maksimal dan masih konvensional, yaitu berupa brosur. Informasi di dalam satu brosur cenderung terbatas dan mudah rusak.

Untuk itulah dibuat sebuah aplikasi yang berisi informasi wisata di Kota Solo. Aplikasi ini bersifat interaktif dimana *user* bisa memilih lokasi mana yang ingin diketahui. Di dalam aplikasi juga disertai peta wisata minimalis yang berisi lokasi wisata. [10] Berdasarkan penelitian sebelumnya mengenai Aplikasi *Web*

Pemetaan Informasi Wisata Sejarah dan Budaya pada Wilayah DKI Jakarta. Dalam pembuatan aplikasi pemetaan informasi budaya DKI Jakarta ini membahas tentang kebudayaan yang ada di daerah DKI Jakarta meliputi informasi Sejarah Jakarta, Cagar Budaya, Wisata Perkampungan Budaya, Wisata Religi, Museum, Wisata Bahari serta Wisata pendukung seperti Taman Kota. Web aplikasi pemetaan wisata sejarah dan budaya DKI Jakarta ini menyajikan peta digital yang didalamnya terdapat informasi mengenai lokasi wisata pada DKI Jakarta diantaranya sejarah, lokasi, keunggulan yang ada.

Pada aplikasi web pemetaan ini dilengkapi dengan informasi pelengkap diantaranya profil, berita-berita mengenai pariwisata, sejarah, masyarakat dan budaya. [10] Berdasarkan uraian terdahulu, dalam penelitian ini akan mengacu pada penelitian tentang aplikasi mobile pemandu objek wisata sejarah dan budaya berbasis web. Untuk tugas akhir ini lebih menekankan potensi budaya daerah yang ada sebagai tujuan untuk membantu meningkatkan pengalaman belajar budaya bagi wisatawan dengan tipe *purposeful cultural tourist*. Sehingga para wisatawan tersebut bisa mendapatkan pengalaman belajar budaya dengan mudah dan nyaman. Untuk teknologi yang digunakan dalam penelitian ini dengan aplikasi *mobile Android* dengan pemetaan dan pengalaman budaya yang ada di tempat tersebut untuk menambah wawasan wisatawan yang datang dalam konteks budaya Jawa.

2.2. Pariwisata Budaya

Pariwisata budaya merupakan jenis pariwisata yang berdasarkan pada mosaik tempat, tradisi, kesenian, upacara-

upacara, dan pengalaman yang memotret bangsa/suku bangsa, yang merefleksikan keanekaragaman dan identitas dari masyarakat atau bangsa bersangkutan [11].

Secara konseptual berdasarkan referensi definisi dan acuan yang ada, pariwisata budaya adalah suatu “*konsep*” pengembangan pariwisata berbasis sumberdaya budaya yang bertujuan untuk mendukung upaya-upaya pelestarian budaya dan lingkungannya. Caranya adalah melalui peningkatan partisipasi masyarakat dalam memanfaatkan secara berkelanjutan sumberdaya budaya sebagai daya tarik pariwisata guna meningkatkan taraf hidup dan ekonomi masyarakat setempat [11].

Setiap orang akan memberikan pengertian yang berbeda, di satu sisi pariwisata budaya diartikan sebagai pariwisata yang berhubungan dengan cipta karya (*creative art*) seperti teater, tari, opera dan lukis. Di sisi lain diartikan sebagai pariwisata yang berhubungan dengan cipta Karsa (*Humanities*) seperti sejarah, tradisi, adat istiadat, bahasa dan sebagainya.

Pariwisata budaya khususnya dalam rangka pelestarian, peran masyarakat lokal sebagai pemilik budaya lebih didahulukan dalam memilah komponen budaya mana yang dapat dimanfaatkan untuk kepentingan pariwisata. Oleh karenanya dalam mengakomodasi kepentingan dimaksud, beberapa kriteria yang perlu diperhatikan dalam mengembangkan pariwisata budaya mencakup [11]:

- a. Wisata budaya adalah kegiatan perjalanan seseorang atau kelompok untuk melihat, meneliti, mengetahui, dan memahami kebudayaan (tradisi, perilaku,

kerajinan, kesenian, dll) masyarakat di suatu tempat dalam waktu tertentu,

- b. Wisata budaya adalah kegiatan perjalanan seseorang atau kelompok untuk melihat, meneliti, mengetahui, dan memahami hal-hal yang berbeda dengan kehidupan sehari-hari (eksotis), yang dilakukan dalam waktu tertentu (sementara).
- c. Wisata budaya adalah kegiatan perjalanan seseorang atau kelompok melihat, meneliti, mengetahui, dan memahami kebudayaan masyarakat di suatu tempat dari waktu ke waktu (bukan hanya kebudayaan yang bersifat tradisional saja melainkan kebudayaan yang sudah dipengaruhi oleh kebudayaan lain)
- d. Wisata budaya berkaitan dengan obyek yang memiliki daya tarik kelokalan, menghasilkan nilai tambah dan manfaat, serta berkelanjutan
- e. Wisata budaya juga berkenaan dengan fasilitas, aksesibilitas, pelaku, modal, dan sistem informasi.

2.3. Pasar Tradisional

Pasar tradisional merupakan aset budaya dan penggerak ekonomi yang selama ini terbukti tahan terhadap hantaman krisis ekonomi. Potensi pasar yang besar terhadap pertumbuhan ekonomi daerah dan menjadi tujuan wisata sudah terbukti, seperti halnya Pasar Belingharjo di Yogyakarta, Pasar Klewer di Solo, dan Pasar Apung di Banjarmasin. Pasar juga berfungsi sebagai tempat bermuaranya produk-produk rakyat di sekitarnya dan merupakan penyedia lapangan kerja bagi masyarakat, hingga saat ini pasar tradisional di seluruh Indonesia mencapai 13.450 unit dan mampu menampung lebih

dari 12.625.000 pedagang (Malono. 2011). Menurut Asosiasi Pedagang Pasar Tradisional Seluruh Indonesia (APPSI) pada tahun 2005 sebanyak 400 kios tutup setiap tahunnya karena kalah saing dengan pasar modern. Menurut wakil menteri perdagangan (Wamendag) pertumbuhan pasar modern sebesar 31,4 persen dan pasar rakyat minus 8,1 persen sehingga pendapatan pasar domestik semakin berkurang [12]. Perkembangan pasar modern bukan penyebab utama redupnya pasar tradisional tetapi karena infrastruktur pasar tradisional semakin memburuk, pengorganisasian para PKL, dan pelaksanaan praktik pengelolaan pasar yang lebih baik.

2.4. Pasar Pabean Surabaya

Pasar pabean merupakan salah satu pasar tradisional yang ada di kota Surabaya. Pasar pabean terletak antara Jl. Kembang Jepun dan Jl. KH Mas Mansur. Tidak ada catatan resmi kapan pasar ini berdiri. Namun menurut catatan resmi UPTD Pasar Surya, pasar ini berdiri pada tahun 1918. Depan pasar ada bangunan bergaya *neo classic* peninggalan Belanda yang saat ini digunakan sebagai kantor pengelola pasar. Pasar ini bersifat dinamis, untuk konsumen yang ingin menikmati ikan segar, pasar paben merupakan pilihan yang tepat karena setiap harinya pasar ini ramai dan merupakan tempat transit berbagai hasil laut dan juga tambak dari berbagai kota seperti Sidoarjo, Gresik, Pasuruan dan Jombang. Selain terkenal dengan hasil ikannya Pasar Pabean dianggap sebagai pasar terbesar di Asia Tenggara [13].



Gambar 2. 1 Transaksi Ikan di Pasar Pabean

Dari segi komoditi, Pasar Pabean beroperasi 24 jam tanpa henti dan lengkap. Pasar Pabean memiliki 2 sektor, dimana sektor timur dan barat. Di sektor timur Pasar Pabean terkenal sebagai pusat perkulakan rempah-rempah dan bahan pangan, sedangkan dilihat pada sektor barat Pasar Pabean terkenal dengan pasar ikan terbesar di Surabaya [13].



Gambar 2. 2 Transaksi Rempah-Rempah dan Bahan Pangan

Pasar Pabean memiliki pembagian wisata belanja, diantaranya adalah Wisata Belanja Sayur, Buah, Ikan, Daging, Garment,

Rempah-Rempah, Kue, Bunga, Elektronik, Rombeng, Alat Manten, Batu, Oleh-Oleh, dan Kuliner. Pasar Pabean memiliki sejarah panjang dan sudah terbiasa oleh dinamika perekonomian yang labil di Indonesia, hal ini diyakini oleh sebagian besar pedagang Pasar Pabean. Pasar Pabean memiliki posisi yang kuat untuk bersaing dengan pasar modern, hal ini dilihat dari harga dan kualitas produk. Dua kekuatan itu yang menjadikan Pasar Pabean tetap menjadi simpul perekonomian di Surabaya [13].

Pasar Pabean selain dijadikan untuk kepentingan belanja juga sebagai wisata Sejarah dan Budaya. Pasar ini berdiri sejak era penjajahan Belanda dan masih banyak bangunan bersejarah yang dapat dikunjungi sekitar Pasar Pabean, hal ini memiliki daya tarik atau nilai tinggi untuk setiap wisatawan lokal atau luar negeri yang berkunjung ke Pasar Pabean. Wisatawan dapat melihat pasar ini sebagai perkampungan Multietnis dimana banyak jejak sejarah dari para saudagar dan pelaut dari berbagai daerah yang pernah singgah di Surabaya. Pasar Pabean memiliki jejak sejarah yang banyak, pengunjung akan bisa berkeliling di sekitar Pasar Pabean disuguhkan dengan bangunan-bangunan kuno peninggalan Belanda, melewati lorong-lorong bangunan bersejarah, Toko penjual yang masih menggunakan gedung tua [13].

2.5. Wisata Surabaya

Surabaya merupakan pelabuhan utama dan pusat perdagangan komersial di wilayah timur Indonesia, dan sekarang menjadi salah satu kota terbesar di Asia Tenggara. Bersama dengan Lamongan di barat laut, Gresik di barat, Bangkalan di timur laut, Sidoarjo di selatan, Mojokerto dan Jombang di barat daya

menjadi kesatuan yang dinamakan Gerbang Kertosusila, seperti Jabodetabek di Jakarta dan sekitarnya [14].

Letak kota Surabaya berada di tepi pantai utara Jawa timur, dan berbatasan langsung dengan Selat Madura di utara dan timur, Kabupaten Sidoarjo di selatan, dan Kabupaten Gresik di barat. Kota Surabaya berada pada dataran rendah, dengan ketinggian antara 3-6 M di atas permukaan laut, hal ini juga yang menyebabkan suhu udara di Surabaya tergolong panas dan kering.

Sebagai kota metropolitan, Surabaya dihuni oleh multi etnis dan banyak suku bangsa, seperti warga Tionghoa, suku Jawa, Batak, Madura, Bali, Bugis, Sunda dan banyak lagi. Ada juga warga negara asing termasuk Malaysia, Cina, India, Arab dan Eropa. Meskipun Surabaya banyak dipengaruhi oleh beragam budaya, tapi keaslian budayanya masih tetap hidup dan berkembang sampai saat ini. Beragam kebudayaan asli itu bisa dilihat dari berbagai acara seperti, kesenian wayang, ludruk, tari remo dan kebudayaan khas Jawa lainnya [14]. Inilah yang membedakan kota Surabaya dengan kota-kota di Indonesia. Bahkan ciri khas ini sangat kental mewarnai kehidupan pergaulan sehari-hari. Sikap pergaulan yang sangat egaliter, terbuka, berterus terang, kritik dan mengkritik merupakan sikap hidup yang dapat ditemui sehari-hari. Bahkan kesenian tradisional dan makanan khasnya mencerminkan pluralisme budaya Surabaya [15].

Bahasa yang digunakan sehari-hari sebagian besar menggunakan bahasa Jawa, dengan dialek Suroboyoan. Dialek ini cukup berbeda dengan bahasa Jawa dari Jawa Tengah seperti Surakarta atau Yogyakarta. Dialek Suroboyoan memiliki intonasi yang dalam dan tinggi, dan terkesan keras.

Dari segi pariwisata, layaknnya kota yang sarat akan sejarah, Surabaya memiliki beberapa obyek wisata yang bisa dikunjungi yang berhubungan dengan sejarah masa lampau. Ditambah lagi, Surabaya memiliki keanekaragaman kuliner yang selalu dicari oleh wisatawan yang datang, seperti: rujak uleg, rawon, nasi bebek, kupang lontong, longtong balap dan masih banyak lagi [14].

2.6. Android

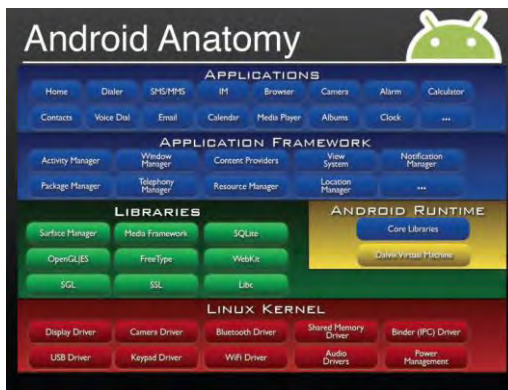
Android adalah sistem operasi yang berbasis Linux untuk telepon seluler seperti telepon pintar dan komputer tablet. Android menyediakan platform terbuka bagi para pengembang untuk menciptakan aplikasi mereka sendiri untuk digunakan oleh bermacam peranti bergerak. Awalnya, Google Inc. membeli Android Inc., pendatang baru yang membuat peranti lunak untuk ponsel. Kemudian untuk mengembangkan Android, dibentuklah Open Handset Alliance, konsorsium dari 34 perusahaan peranti keras, peranti lunak, dan telekomunikasi, termasuk Google, HTC, Intel, Motorola, Qualcomm, T-Mobile, dan Nvidia [16]

Pada saat perilisn perdana Android, 5 November 2007, Android bersama Open Handset Alliance menyatakan mendukung pengembangan standar terbuka pada perangkat seluler. Di lain pihak, Google merilis kode-kode Android di bawah lisensi Apache, sebuah lisensi perangkat lunak dan standar terbuka perangkat seluler.

Di dunia ini terdapat dua jenis distributor sistem operasi Android. Pertama yang mendapat dukungan penuh dari Google atau *Google Mail Services* (GMS) dan kedua adalah yang benar-benar bebas distribusinya tanpa dukungan

langsung Google atau dikenal sebagai Open Handset Distribution (OHD).

Dalam paket sistem operasi Android terdiri dari beberapa unsur seperti tampak pada gambar 2.3, secara sederhana arsitektur Android merupakan sebuah kernel Linux dan sekumpulan pustaka C / C++ dalam suatu framework yang menyediakan dan mengatur alur proses aplikasi [16].



Gambar 2. 3 Android Anatomy

2.6.1. Android SDK (Software Development Kit)

Android SDK adalah tools API (Application Programming Interface) yang diperlukan untuk memulai pengembangan aplikasi dengan platform Android menggunakan bahasa pemrograman Java. Android merupakan subset perangkat lunak untuk ponsel yang meliputi sistem operasi, middleware dan aplikasi kunci yang dirilis oleh Google. Sampai saat ini Google menyediakan Android SDK (*Software Development Kit*) sebagai alat bantu dan API untuk memulia pengembangan aplikasi Android. Sebagai platform aplikasi-netral, Android memberi semua orang kesempatan untuk membuat aplikasi yang dibutuhkan [17].

Beberapa fitur paling penting pada sebuah perangkat Android diantaranya :

- 1) *Framework* Aplikasi yang mendukung penggantian komponen dan *reusable*.
- 2) Mesin *Virtual Dalvik* dioptimalkan untuk perangkat *mobile*.
- 3) *Integrated browser* berdasarkan *engine open source Webkit*.
- 4) Grafis yang dioptimalkan dan didukung oleh *libraries* grafis 2D, grafis 3D, berdasarkan spesifikasi ES 1.0 (Opsional akselerasi *hardware*).
- 5) *SQLite* yang sudah terintegrasi untuk penyimpanan data.
- 6) Media Support yang mendukung audio, video dan gambar.
- 7) Kamera, GPS dan *Accelerometer*.
- 8) Lingkungan *development* yang lengkap termasuk perangkat *emulator*, *tool* untuk *debugging*, profil kinerja memori dan plugin untuk IDE Eclipse.

2.6.2. ADT (Android Development Tools)

Android Development Tools (ADT) adalah plugin yang didesain untuk IDE Eclipse yang memberikan kemudahan dalam mengembangkan aplikasi untuk Android. Dengan menggunakan ADT untuk Eclipse bisa memudahkan dalam membuat struktur aplikasi, membuat GUI aplikasi, dan menambahkan komponen-komponen yang lainnya. Begitu juga dengan ADT untuk Eclipse ini, aplikasi dapat di running menggunakan Android Virtual Device yang sudah bundle dengan ADT ketika diinstal pada Eclipse. ADT ini juga memungkinkan Eclipse untuk membuat package aplikasi android (.apk) yang digunakan untuk mendistribusikan aplikasi yang telah dibuat. Untuk instalasi ADT pada Eclipse dapat dilakukan secara offline dan online. Paket ADT ini bisa diunduh di situs Developer Android [18].

2.6.3. Eclipse

Eclipse adalah sebuah IDE (*Integrated Development Environment*) untuk mengembangkan perangkat lunak dan dapat dijalankan di semua platform (platform-independent). Eclipse merupakan satu-satunya IDE yang disarankan oleh Google untuk pembuatan aplikasi berbasis Android. Versi Eclipse yang digunakan adalah versi Juno, karena pada saat pembuatan aplikasi versi paling baru adalah Eclipse Juno. Berikut ini adalah sifat dari IDE Eclipse:

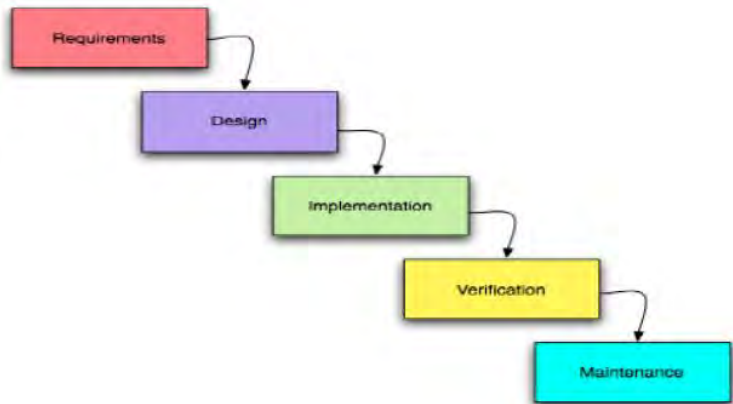
- 1) Multi-platform, yaitu Target sistem operasi Eclipse adalah Microsoft Windows, Linux, Solaris, dan Mac OS X.
- 2) Multi-language, Eclipse dikembangkan dengan bahasa pemrograman Java, akan tetapi Eclipse mendukung pengembangan aplikasi berbasis bahasa pemrograman lainnya, seperti C/C++, Python, PHP, dan lain sebagainya.
- 3) Multi-role, Selain sebagai IDE untuk pengembangan aplikasi, Eclipse pun bisa digunakan untuk aktivitas dalam siklus pengembangan perangkat lunak, seperti dokumentasi, test perangkat lunak, pengembangan web, dan lain sebagainya.

Eclipse pada saat ini merupakan salah satu IDE favorit dikarenakan gratis dan open source, yang berarti setiap orang boleh melihat kode pemrograman perangkat lunak ini. Selain itu, kelebihan dari Eclipse yang membuatnya populer adalah kemampuannya untuk dapat dikembangkan oleh pengguna dengan komponen yang dinamakan plug-in.

2.7. Iterative Waterfall Model

Model waterfall adalah proses pengembangan perangkat lunak tradisional yang umum digunakan dalam proyek-proyek perangkat lunak yang paling pembangunan. Ini adalah model

sekuensial, sehingga penyelesaian satu set kegiatan menyebabkan dimulainya aktivitas berikutnya. Hal ini disebut waterfall karena proses mengalir "secara sistematis dari satu tahap ke tahap lainnya dalam mode ke bawah [19]". Membentuk kerangka kerja untuk pengembangan perangkat lunak. Beberapa varian dari model ada, setiap label yang berbeda menggunakan untuk setiap tahap. Secara umum, bagaimanapun, model ini dianggap memiliki enam tahap yang berbeda seperti yang ditunjukkan pada Gambar 2.4 yaitu: analisis Kebutuhan, desain, implementasi, verifikasi, instalasi dan pemeliharaan [20].



Gambar 2. 4 Waterfall Model

2.8. XML

XML (*eXtensible Markup Language*) merupakan bahasa *web* turunan dari SGML (*Standart Generalized Markup Language*) yang ada sebelumnya. XML hampir sama dengan HTML, dimana keduanya sama-sama turunan dari SGML. Teknologi XML dikembangkan mulai tahun 1966 dan mendapatkan pengakuan dari *Worl Wide Web Consortium* (W3C) pada

bulan Februari 1998. Sedangkan SGML sendiri telah dikembangkan pada awal tahun 1980 an. Pada saat HTML dikembangkan pada tahun 1990, para penggagas XML mengadopsi bagian paling penting SGML dan dengan berpedoman pada pengembangan HTML menghasilkan bahasa *markup* yang tidak kalah hebatnya dengan SGML. Secara sederhana XML adalah suatu bahasa yang digunakan untuk mendeskripsikan dan memanipulasi dokumen secara terstruktur. Secara teknis XML didefinisikan sebagai suatu bahasa *meta-markup* yang menyediakan format tertentu untuk dokumen-dokumen yang mempunyai data terstruktur.

2.8.1. XML untuk Android

User Interface (UI) di Android dapat dibangun dengan dua cara, yakni dengan mendefinisikan XML atau dengan menulis kode Java. Mendefinisikan struktur GUI dalam XML lebih banyak digunakan, karena sebagai salah satu dari prinsip *Model-Viewer-Control*, dimana UI haruslah selalu terpisah dari logika program. Tata letak yang berbasis XML adalah spesifikasi hubungan *widget* satu sama lain dan kontainer dalam format XML. Android menjadikan XML sebagai basis layout untuk menjadikannya *resources*, dan disimpan dalam *res/layout* direktori dalam *project* android.

Dalam pembuatan antar muka dikenal istilah *resource*. *Resources* adalah file eksternal (non-kode file) yang digunakan oleh penulis kode dan dikompilasi pada saat membangun aplikasi. Android mendukung beberapa jenis file yang berbeda sebagai *resources*, yakni XML, PNG, dan file JPEG [18]

BAB III METODOLOGI

Bagian ini menjelaskan rangkaian dari tahapan-tahapan yang dilakukan dalam pengerjaan Tugas Akhir ini. Rangkaian pengerjaan tugas akhir ini mengacu pada model pengembangan perangkat lunak *waterfall* sebagaimana digambarkan pada Gambar 3.1 di bawah ini:



Gambar 3. 1 Tahapan Pengerjaan Tugas Akhir

3.1. Tahapan Pelaksanaan Tugas Akhir

Pada bab ini, akan dijelaskan tentang Studi Literatur, Identifikasi Kebutuhan Perangkat Lunak, Perancangan Aplikasi, Pengkodean Aplikasi, Pengujian Aplikasi, Penyusunan Laporan Tugas Akhir, Jadwal Pelaksanaan.

3.1.1. Studi Literatur

Studi literatur yang dilakukan antara lain pembelajaran dan pemahaman literatur mengenai wisata budaya, wisata type *purposeful cultural tourist* dan konsep-konsep dalam dunia pariwisata dan akomodasi. Hal lain yang akan dipelajari adalah pemahaman mengenai *Android* dan komponen pendukung lain dalam pengembangan aplikasi.

3.1.2. Identifikasi Kebutuhan Perangkat Lunak

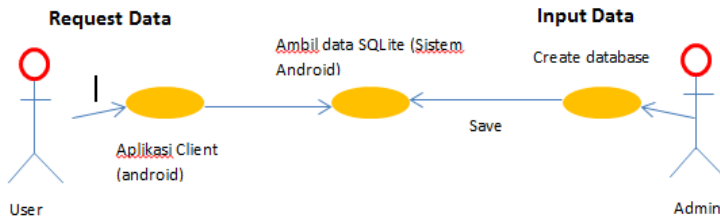
Pada tahap ini akan dilakukan analisa mengenai kebutuhan dari layanan pemandu pengalaman kultural dalam konteks budaya berdasarkan studi kasus lain yang sudah ada serta melalui studi literatur. Tahapan ini nantinya akan menghasilkan kebutuhan fungsional dan non-fungsional dari perangkat lunak.

3.1.3. Perancangan Aplikasi

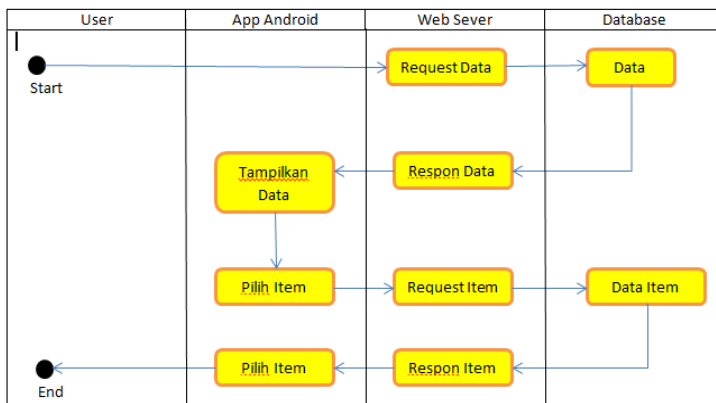
Setelah melakukan identifikasi kebutuhan, tahap selanjutnya adalah merancang aplikasi pemandu pengalaman kultural. Dalam tahap ini, *use case* dibuat berdasarkan kebutuhan perangkat lunak yang sudah didefinisikan sebelumnya dan dilanjutkan dengan pembuatan *activity diagram* dan *sequence diagram*.

Untuk alurnya dapat dilihat pada gambar 3.2 dimulai ketika *user* mulai menjalankan aplikasi, aplikasi tersebut mengambil data dari basis data Aplikasi *Cultural Tourism Guide* melalui

SQLite. Data yang diambil adalah data pada *basis data* yang telah disimpan di sistem android(*SQLite*). Data tersebut lalu ditampilkan ke dalam aplikasi berdasarkan menu yang dipilih user.



Gambar 3. 2 Use Case Diagram

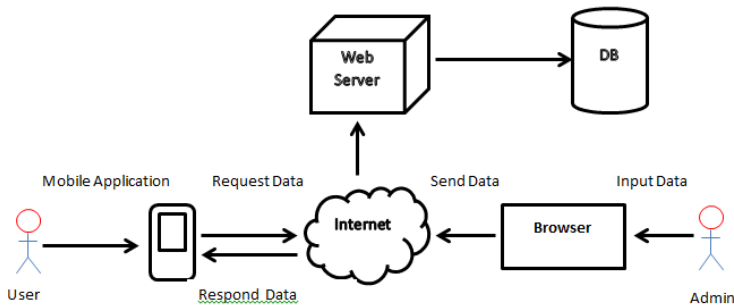


Gambar 3. 3 Activity Diagram User

Pada Gambar 3.3 diatas dijelaskan activity diagram user pada aplikasi android. Pada saat pertama aplikasi dijalankan user akan membuka menu kategori. Daftar kategori ini ditampilkan setelah melakukan koneksi ke web server untuk mengambil data kategori pada basis data Aplikasi *Cultural Tourism Guide*

dan menampilkan kembali ke menu kategori pada aplikasi android. Setelah user memilih salah satu dari daftar kategori yang sudah ditampilkan maka akan dibuka menu item berdasarkan item yang telah dipilih.

Pada Gambar 3.4 merupakan gambar arsitektur sistem keseluruhan pada aplikasi. Pada gambar tersebut dijelaskan, user menggunakan aplikasi Android dan internet untuk melakukan request data pada web server. Web server melanjutkan untuk mengambil data pada basis data. Admin menggunakan browser dan internet untuk memasukkan data ke dalam web server. Data yang diterima oleh web server akan disimpan di dalam basis data server.



Gambar 3. 4 Arsitektur Sistem

3.1.4. Pengkodean Aplikasi

Pada tahap ini, akan dimulai proses *coding* atau pengkodean dari aplikasi berdasarkan desain yang telah dibuat sebelumnya. Pada tugas akhir ini, bahasa pemrograman yang digunakan adalah *Java* serta menggunakan *database* MySQL.

3.1.5. Pengujian Aplikasi

Tahapan ini dilakukan setelah pengkodean aplikasi telah selesai. Pengujian aplikasi dilakukan untuk mengecek apakah aplikasi sudah berjalan dengan baik atau belum. Apabila masih ada beberapa komponen yang belum sempurna atau tidak memenuhi kriteria pengujian maka perbaikan akan dilakukan sampai perangkat lunak dapat berfungsi dengan baik.

3.1.6. Penyusunan Laporan Tugas Akhir

Tahapan ini merupakan bagian dokumentasi proses pengembangan perangkat lunak dari mulai identifikasi kebutuhan sampai dengan pengujian aplikasi. Dokumentasi tersebut akan dituangkan ke dalam laporan tugas akhir.

3.1.7. Jadwal Pelaksanaan

Penentuan jadwal kegiatan di dalam penelitian sangatlah penting untuk memberikan acuan dalam mengerjakan tugas akhir ini. Sehingga semua kegiatan terencana dengan baik dan selesai tepat waktu dengan hasil sesuai yang diharapkan.

Pengerjaan tugas akhir ini diharapkan selesai dalam waktu 4 bulan, yang dimulai pada bulan September- Desember. Untuk lengkapnya bisa dilihat pada



Gambar 3. 5 Jadwal Pelaksanaan Pengerjaan Tugas Akhir.

BAB IV PERANCANGAN

Pada bab ini akan dijelaskan mengenai rancangan pengembangan purwarupa aplikasi pemandu pengalaman kultural dalam konteks budaya (*mobile application for cultural experience guide in the Javanese context*).

4.1. Kebutuhan Fungsional

Pada tahap ini akan dilakukan analisa mengenai kebutuhan dari layanan pemandu pengalaman kultural dalam konteks budaya berdasarkan studi kasus lain yang sudah ada serta melalui studi literatur. Tahapan ini nantinya akan menghasilkan kebutuhan fungsional dari perangkat lunak seperti pada Tabel 4. 1

Tabel 4. 1 Daftar kebutuhan fungsional

ID FR	Deskripsi Kebutuhan Fungsional
FR01	Wisatawan dapat melihat informasi produk dan pedagang yang diinginkan.
FR02	Melihat informasi penting terkait dengan tindakan kriminal.
FR03	Sistem memberikan informasi lokasi yang diinginkan
FR04	Memberikan informasi apa saja yang harus diperhatikan saat berkunjung ke pasar pabean.
FR05	Sistem menyediakan istilah-istilah atau bahasa yang digunakan di pasar pabean
FR06	Sistem menyediakan informasi harga komoditi terupdate yang sinkron dengan data harga

ID FR	Deskripsi Kebutuhan Fungsional
	produk di Pasar Pabean Surabaya.
FR07	Sistem menyediakan informasi suku dan kebiasaan pedagang di Pasar Pabean Surabaya.

4.2. Aktor-Aktor

Berdasarkan dari hasil analisa kebutuhan fungsional dapat dipetakan aktor-aktor yang terlibat dalam aplikasi *culture experience* yang dikembangkan. Rincian informasi mengenai aktor-aktor yang terlibat akan dijabarkan pada Tabel 4. 2

Tabel 4. 2 Daftar Aktor

Aktor	Deskripsi
Wisatawan	Orang yang menggunakan aplikasi untuk memilih informasi yang diinginkan saat berkunjung ke Pasar Pabean.

4.3. Use Case

Use case dibuat dengan tujuan untuk memenuhi kebutuhan aplikasi yang sudah dibuat pada bab sebelumnya. Daftar use case yang dibuat dapat dilihat pada Tabel 4. 3.

Tabel 4. 3 Daftar Use Case

Use Case ID	Use Case	Aktor
UC01-01	Mencari informasi penjual, barang dagangan yang ingin dikunjungi	Wisatawan
UC02-01	Melihat informasi penting terkait tindakan kriminal	Wisatawan

Use Case ID	Use Case	Aktor
UC03-01	Melihat dan mencari lokasi yang diinginkan di pasar pabean.	
UC04-01	Melihat rincian informasi terkait budaya di Pasar Pabean Surabaya.	Wisatawan
UC05-01	Melihat informasi terkait istilah-istilah dan ekspresi-ekspresi bahasa yang sesuai dengan budaya setempat.	Wisatawan
UC06-01	Melihat rincian informasi harga tiap produk	Wisatawan
UC07-01	Melihat informasi suku dan kebiasaan secara garis besar untuk pedagang.	Wisatawan

4.4. Activity Diagram

Activity diagram dibuat berdasarkan dari *use case* yang ada. *Activity diagram* bertujuan untuk menjelaskan alur kerja use case. Daftar *activity diagram* yang telah dibuat dapat dilihat di Tabel 4. 4

Tabel 4. 4 Daftar Aktivitas

UC ID	AC ID	Aktivitas
UC01-1	AC01-1-01	Mencari informasi barang dagangan yang diinginkan
	AC01-1-02	Mencari informasi pedagang berdasarkan produk yang dipilih.
UC02-1	AC02-1-01	Melihat informasi tindakan yang sering terjadi di pasar, beberapa modus pelaku.
UC03-1	AC03-1-01	Melihat titik lokasi penjual di peta.
	AC03-1-02	Memberikan petunjuk arah pada peta ketika ingin

UC ID	AC ID	Aktivitas
		berkunjung ke pasar pabean dari tempat atau lokasi yang diinginkan.
UC04-1	AC04-1-01	Memberikan informasi etika saat berada di pasar, bagaimana melakukan transaksi jual beli yang baik.
UC05-1	AC05-1-01	Melihat informasi terkait istilah-istilah dan ekspresi-ekspresi bahasa jawa dengan dialek Surabaya di pasar.
	AC05-1-02	Melihat informasi ekspresi-ekspresi bahasa jawa dengan dialek surabaya di pasar yang dilakukan penjual dan pembeli.
	AC05-1-03	Melihat informasi ekspresi-ekspresi angka Jawa Surabaya.
UC06-1	AC06-1-01	Mencari dan melihat informasi harga komoditi terkini di Pasar Pabean Surabaya.
UC07-1	AC07-1-01	Melihat informasi suku dan kebiasaan pedagang di Pasar Pabean Surabaya.

4.5. Sequence Diagram

Sequence diagram menampilkan alur dari *activity diagram* dengan penjelasan yang berisi metode / prosedur pada pemrograman aplikasi. *Sequence diagram* bisa dilihat di lampiran C

4.6. Class Diagram

Class diagram berisi gambaran desain obyek yang akan dibuat pada tahap pengkodean aplikasi. Pembuatan *class*

diagram bertujuan untuk memberikan gambaran obyek yang lebih detail yang berisi *behaviour* dan *attribute* dari objek untuk memudahkan dalam melakukan pengkodean aplikasi. Untuk *detail class diagram* bisa dilihat pada lampiran D.

4.7. Entity Relationship Diagram (ERD)

Diagram *Entity Relationship Diagram* (ERD) untuk pengembangan purwarupa aplikasi dapat dilihat pada lampiran E

4.8. Penggunaan Test Case

Test Case merupakan bentuk pengujian terhadap setiap use case yang telah dibuat. *Test Case* berfungsi mengecek apakah setiap *use case* dalam aplikasi dapat berjalan dengan baik.

(halaman ini sengaja dikosongkan)

BAB V IMPLEMENTASI

5.1. Lingkungan Implementasi

Aplikasi SQL *Battle* ini dikembangkan dengan menggunakan perangkat keras dengan spesifikasi yang dapat dilihat pada Tabel 5. 1. Sedangkan untuk perangkat lunak, aplikasi ini dikembangkan dengan menggunakan beberapa teknologi terkait dengan spesifikasi yang dapat dilihat pada Tabel 5. 2

Tabel 5. 1 Spesifikasi Perangkat Keras

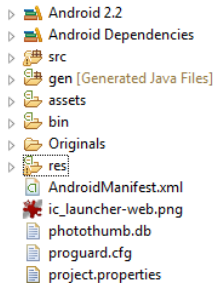
Aplikasi	Perangkat Keras	Spesifikasi
<i>Cultural Experience Guide</i>	NoteBook	Prosesor : AMD E2-1800 APU with Radeon (1.70GHz)
		Memori : 4 GB
		Sistem Operasi : Windows 7 Ultimate (64Bit)

Tabel 5. 2 Spesifikasi Perangkat Lunak

Aplikasi	Perangkat Lunak	Spesifikasi
<i>Cultural Experience Guide</i>	Web Server	Apache 2.4.10
	Database	1.) SQLite 0.8.3.1 2.) MySQL 5.1.61
	Bahasa Pemrograman	1.) PHP 5.5.11 2.) Java
	Framework Pemrograman	Codeigniter 3.0
	Editor	Notepad ++

5.2. Struktur Direktori

Struktur direktori dari aplikasi ini mengikuti struktur direktori dari *Java Project (Eclipse)*. Struktur direktori aplikasi ini bisa dilihat pada Gambar 5. 1



Gambar 5. 1 Struktur Direktori Project

5.3. Implementasi Aplikasi Cultural Experience Guide

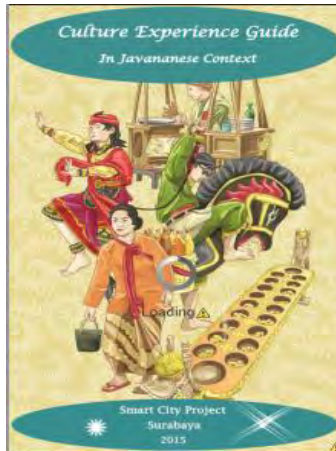
Pada bab ini, akan dijelaskan tentang Antar Muka Halaman Utama, Kode Program Halaman Utama, Mencari Informasi Produk, Melihat Titik Lokasi Pedagang, Memberikan Informasi Etika di Pasar, Memberikan Informasi Dialek Jawa di Pasar, Memberikan Informasi Dialek Pasar Antara Pedagang dan Pembeli, Melihat dan Menentukan Informasi Petunjuk Arah.

5.3.1. Antar Muka Halaman Utama

Berdasarkan perancangan, halaman menu utama merupakan kumpulan menu–menu utama aplikasi yang terhubung dengan beberapa class pendukung. Halaman menu utama merupakan interface yang berfungsi sebagai interaksi user dengan semua menu utama yang terdapat pada aplikasi. Pada halaman menu utama tersebut dihubungkan dengan tiga bentuk *interface* untuk mengumpulkan menu berdasarkan fungsinya. Pembuatan tampilan antarmuka pada halaman menu utama di implementasikan dalam bentuk XML. Setiap elemen dalam

tampilan antarmuka perlu ditambahkan atribut pengenalan, sehingga elemen tersebut akan di *generate* dalam atribut XML dan memudahkan untuk digunakan pada kelas yang memerlukan.

Interface pertama yang dibuat adalah halaman *splashscreen.xml*. Didalamnya terdapat dua atribut yaitu *ProgressBar* dan *TextView*. Untuk mengatur tata letak *layout* agar setiap atribut tidak berantakan diperlukan *LinearLayout*. Atribut didalamnya antara lain *layout_width* untuk mengukur lebar isi *widget*, *layout_height* untuk mengatur tinggi *widget*, *background* untuk memberikan gambar latar belakang dari *layout* atau *widget*, *gravity* untuk mengatur perataan *widget* atau isi teks, dan *orientation* untuk mengatur posisi *widget* apakah digunakan sebagai *vertical* atau *horizontal*. Untuk tampilan *splashscreen* dapat dilihat pada Gambar 5. 2



Gambar 5. 2 Tampilan layout splashscreen

Interface kedua yang dibuat adalah halaman *rinci.xml*. Didalamnya terdapat tiga atribut yaitu *ImageView*, *Button* dan

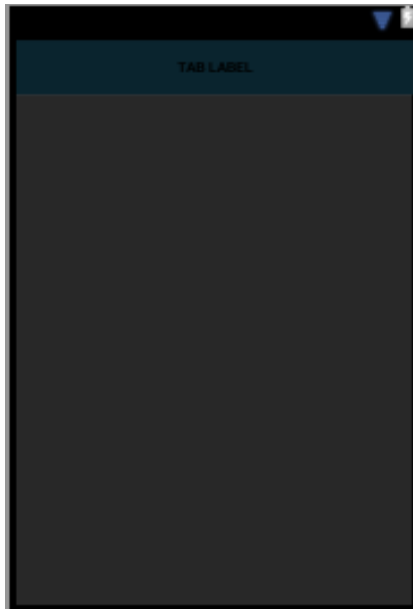
TextView. Untuk *interface* yang kedua menggunakan dua *LinearLayout*, hal ini dimaksudkan agar setiap atribut yang digunakan tidak berantakan. Untuk atributnya lainnya sudah dijelaskan pada halaman sebelumnya. Untuk tampilan layout rinci dapat dilihat pada Gambar 5. 3.



Gambar 5. 3 Tampilan layout rinci

Interface ketiga yang dibuat adalah halaman *main.xml*. Pada *tab* menu menggunakan *layout TabHost* yang didalamnya terdapat atribut *FrameLayout* dan *TabWidget* sebagai atribut pembangun *TabHost*. *TabWidget* sendiri digunakan untuk menampilkan *tab* menu dari program yang dibuat.

Untuk hasil tampilan dari *layout* main dapat dilihat pada Gambar 5. 4.



Gambar 5. 4 Tampilan *layout* main

5.3.2. Kode Progam Halaman Utama

File class java merepresentasikan layout XML yang telah dibangun menjadi sebuah interface. Penulisan program java membuat fungsi – fungsi dari atribut XML menjadi sebuah antar muka yang mempunyai event / fungsi ataupun tidak.

Halaman Utama melibatkan beberapa *class* diantaranya adalah MainActivity, rinci, dan SplashScreen. *Class* MainActivity mempresentasikan layout main.xml. Kode didalamnya membangun tab menu dan menentukan beberapa tab yang akan dibuat beserta intent menuju *class* lain ketika tab dipilih.

Untuk potongan kode programnya dapat dilihat pada Gambar 5.5.

```
public class MainActivity extends TabActivity
{
    /** Called when the activity is first
    created. */
    private AlertDialog.Builder builder;
    private AlertDialog aboutDialog;
    final String[] tab_menu = {
        "Main Menu",
        "Produk",
        "Pedagang"};

    @Override
    public void onCreate(Bundle
savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.main);

        Resources res = getResources(); //
Resource object to get Drawables
        TabHost tabHost = getTabHost(); //
The activity TabHost
        TabHost.TabSpec spec; // Reusable
TabSpec for each tab
        Intent intent; // Reusable Intent for
each tab

        // All
        intent = new Intent().setClass(this,
rinci.class);
        intent.putExtra("DICT", 1);
        spec =
tabHost.newTabSpec("all").setIndicator(tab_men
u[0],
res.getDrawable(R.drawable.tab_style)).setCont
ent(intent);
        tabHost.addTab(spec);
    }
}
```

Gambar 5.5 Potongan kode class rinci

Selanjutnya untuk menjalankan semua fungsi yang ada di *layout* rinci.xml memerlukan *class* rinci.java, didalamnya

terdapat beberapa *Intent* untuk menuju *class* ketika tombol *button* dipilih. Untuk method yang digunakan dapat dilihat pada Gambar 5. 6

```
public class rinci extends Activity {
    Button p ,h,t, m,d,r;
    Intent translate_dialog;
    private int DICT = 1;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.rinci);

        /* UI Initialization */
        p = (Button) findViewById(R.id.dialek);
        h = (Button) findViewById(R.id.cproduk);
        t = (Button) findViewById(R.id.cpedagang);
        m = (Button) findViewById(R.id.maps);
        d = (Button) findViewById(R.id.bantuan);
        r = (Button) findViewById(R.id.more);

        d.setOnClickListener(new OnClickListener() {

            public void onClick(View v) {
                translate("Cultural Experience Guide");
            }

        });
    }
}
```

Gambar 5. 6 Class rinci dan method memanggil class lain.

Terakhir untuk menampilkan halaman utama saat pertama aplikasi di *load* adalah *Layout* SplashScreen.xml membutuhkan *class* SplashScreen.Java. Didalamnya terdapat beberapa atribut, *class*, metod. Db_kamus_helper untuk memanggil database yang digunakan dan menginisialisasi database yang ada di folder asset, database yang digunakan adalah database SQLite. Untuk memanggil *class* lain membutuhkan metode public void InitDatabase, setelah berhasil menginisialisasi database yang digunakan nantinya baru akan memanggil *class* lain yang sudah ditentukan. Untuk potongan kode Splashscreen.java dapat dilihat pada Gambar 5. 7.

```

public class SplashScreen extends Activity {

    ProgressBar progressBar;
    TextView state;
    Handler handler = new Handler();
    DBKamus db_kamus_helper;

    @Override
    public void onCreate(Bundle savedInstanceState){
        super.onCreate(savedInstanceState);
        setContentView(R.layout.splash_screen);
        db_kamus_helper = new DBKamus(SplashScreen.this);
        state = (TextView) findViewById(R.id.state);
        InitDatabase();
    }
}

```

Gambar 5. 7 Method menentukan database.

Untuk memanggil *class* lainnya dibutuhkan method `InitDatabase`, setelah install database ke aplikasi, *class* lain baru bisa dipanggil. Untuk potongan kodenya dapat dilihat pada Gambar 5. 8.

```

public void InitDatabase() {
    new Thread(new Runnable() {
        @Override
        public void run() {
            delay(1200);
            if (!db_kamus_helper.isDatabaseExist()) {
                handler.post(new Runnable() {
                    @Override
                    public void run() {
                        state.setText("Menginstal Database ...");
                    }
                });
                try {db_kamus_helper.createDataBase();} catch (IOException e) {e.printStackTrace();}
            }
            handler.post(new Runnable() {
                @Override
                public void run() {
                    state.setText("Memulai ...");
                }
            });
            delay(800);
            handler.post(new Runnable() {
                @Override
                public void run() {
                    // TODO Auto-generated method stub
                    Intent main_intent = new Intent(SplashScreen.this, MainActivity.class);
                    SplashScreen.this.startActivity(main_intent);
                    SplashScreen.this.finish();
                }
            });
        }
    });
}

```

Gambar 5. 8 Potongan kode method `InitDatabase Class` `Splashscreen`.

5.3.3. Mencari Informasi Produk

Pencarian informasi produk apa saja yang ada di Pasar Pabean Surabaya dapat diakses dengan mengklik tombol “produk” yang ada di halaman utama aplikasi. Untuk menggunakan fitur ini melibatkan dua *class* dan dua *layout*. Yang pertama adalah *class* rinci.java. *Class* ini memanggil *class* Arti.java ketika user mengklik tombol “Produk”. Method untuk action ketika button daftar produk di klik. Untuk metodenya dapat dilihat pada Gambar 5. 9

```
h.setOnClickListener(new OnClickListener() {
    @Override
    public void onClick(View v) {
        translate("Daftar Produk");
    }
});
```

Gambar 5. 9 Method Action button klik daftar produk

Untuk memanggil *class* lain menggunakan *method* translate. Untuk potongan kodenya dapat dilihat pada Gambar 5. 10

```
protected void translate(String text) {
    translate_dialog = new Intent().setClass(this, Arti.class);
    translate_dialog.putExtra("DICT", DICT);
    translate_dialog.putExtra("TEXT", text);
    startActivity(translate_dialog);
}
```

Gambar 5. 10 Potongan kode rinci.java memanggil class Arti.java.

Selanjutnya untuk mencari informasi produk *class* rinci.java akan memanggil *class* Arti.java. Menggunakan method Boolean untuk menampilkan hasil sesuai dengan data yang ada di database. Didalamnya terdapat bahasa SQL untuk menampilkan data dari *action button* yang diklik oleh user.

Untuk lengkap potongan kodenya dapat dilihat pada Gambar 5. 11.

```
private boolean checkResult() {
    // TODO Auto-generated method stub
    String kingdom1;
    switch(DICT) {
        default:
            case SEMUA:
                kingdom1 = "SELECT * FROM MainMenu where lower(Type)=lower('"+text.replace("'", "''")+ "') ";
                break;
            case HEWAN:
                kingdom1 = "SELECT * FROM Produk where lower>Nama_Produk)=lower('"+text.replace("'", "''")+ "')";
                break;
            case TUMBUHAN:
                kingdom1 = "SELECT * FROM Pedagang where lower(ID_Pedagang)=lower('"+text.replace("'", "''")+ "')";
                break;
    }
    kingdom = db.rawQuery(kingdom1, null);
    if (!kingdom.moveToFirst()) {
        Toast.makeText(this, "word '"+text+"' not found in "+dict_name[DICT-1]+" dictionary !", Toast.LENGTH_SHORT);
        this.finish();
        return false;
    } else return true;
}
```

Gambar 5. 11 Potongan kode Arti.java menampilkan data yang ada di database.

Kode program untuk menampilkan informasi produk di *class* Arti.java terletak pada kode berikut:

```
private boolean checkResult() {
    // TODO Auto-generated method stub
    String kingdom1;
    switch(DICT) {
        default:
            case SEMUA:
                kingdom1 = "SELECT * FROM
MainMenu where
lower(Type)=lower('"+text.replace("'",
"''")+ "') ";
                break;

/* Display Result(Menampilkan hasil ke layout)
*/
        title.setText(text);

content.setTextSize(TypedValue.COMPLEX_UNIT_SP
, font_size);
        switch(DICT) {
```

```

        case SEMUA :
            if (kingdom.getString(3).equals("-
"))){
pic.loadUrl("file:///android_asset/img/Q.jpg");
;

pic.getSettings().setJavaScriptEnabled(true);

        }else{

pic.loadUrl(kingdom.getString(3));
        }

content.setText("Penjelasan\t:\t"+kingdom.getSt
tring(2));

content.setLinkTextColor(Color.rgb(0x44, 0x44,
0xBB));

        break;

```

Gambar 5. 10 Potongan kode Arti.java informasi produk

Kode Arti.java selanjutnya akan menampilkan fungsi dari *method* yang telah dibuat ke halaman atau di android disebut *layout*, yakni menggunakan *layout* arti.xml. Untuk tampilan halaman(*layout*) arti dapat dilihat pada



Gambar 5. 12 Tampilan *layout* arti.xml

Untuk mengetahui *detail* masing-masing produk pengguna masuk ke *Tab Menu* Produk untuk input produk yang ingin dicari. Method didalamnya terdapat SQL untuk *excute* pencarian data produk di dalam *database*. Potongan kodenya dapat dilihat pada

```
protected void cari(String text) {
    // TODO Auto-generated method stub
    list_result.clear();
    if (text.length() > 0) {
        String query;
        switch(DICT) {
            default:
                case HEWAN:
                    query = "SELECT
Nama_Produk FROM Produk where Nama_Produk LIKE
'" + text.replace("'", "''") + "%' ORDER BY
Nama_Produk LIMIT " + search_limit;
                    break;
```

Gambar 5. 13 Potongan kode pencarian produk

Dari data hasil pencarian yang dipilih oleh pengguna maka untuk menampilkannya diperlukan lagi *class* Arti.java. Untuk menampilkan *detail* dari data yang sudah dipilih oleh pengguna ada beberapa kode yang ditulis yakni kingdom1 adalah untuk merepresentasikan pencarian produk dengan beberapa kriteria tertentu menggunakan bahasa SQL. Sedangkan untuk mendapatkan *value* dari *database* untuk ditampilkan ke aplikasi membutuhkan kode Kingdom.getString().

Sedangkan kode content.setLinkTextColor (Color.rgb()); adalah untuk memberikan warna text yang ditampilkan di interface aplikasi. Untuk potongan kodenya dapat dilihat pada Gambar 5. 14

```

if(kingdom.getString(9).equals("-")){
    pic.loadUrl("file:///android_asset/img/Q.jpg");
    pic.getSettings().setJavaScriptEnabled(true);
}else{
    pic.loadUrl(kingdom.getString(9));
}

detail.setText(kingdom.getString(10));
content.setText("Dialek Pasar\t:\t"+kingdom.getString(8)+
    "\nID Pedagang\t:"+kingdom.getString(2)+
    "\nJenis Produk\t:"+kingdom.getString(3)+
    "\nPenjualan Per Hari(Kg)\t:\t"+kingdom.getString(4)+
    "\nPenjualan Jenis Produk(%)\t:\t"+kingdom.getString(5)+
    "\nKategori Produk\t:"+kingdom.getString(6)+
    "\nCiri Produk Terbaik\t:\t"+kingdom.getString(7));
content.setLinkTextColor(Color.rgb(0x44, 0x44, 0xBB));
break;

```

Gambar 5. 14 Potongan kode halaman detail produk

Pencarian informasi penjual/pedagang apa saja yang ada di Pasar Pabean Surabaya dapat diakses dengan mengklik *Tab menu* Pedagang yang ada di halaman utama aplikasi. Untuk menggunakan fitur ini melibatkan tiga *class*. Yang pertama adalah *class* MainActivity.java. Fugsi kode Tabhost dan Intent yang ada didalamnya adalah untuk menampilkan tab menu dan memanggil *class class* yang berhubungan dengan *class* MainActivity. Untuk kode progam *class* Main Activity dapa dilihat pada Gambar 5. 15.

```

// All
intent = new Intent().setClass(this, rinci.class);
intent.putExtra("DICT", 1);
spec = tabHost.newTabSpec("all").setIndicator(tab_menu[0], res.getDrawable(R.drawable.tab_style)).setContent(
tabHost.addTab(spec);

// English to Indonesian
intent = new Intent().setClass(this, Cari.class);
intent.putExtra("DICT", 2);
spec = tabHost.newTabSpec("enid").setIndicator(tab_menu[1], res.getDrawable(R.drawable.tab_style)).setContent(
tabHost.addTab(spec);

// Indonesian to English
intent = new Intent().setClass(this, Cari.class);
intent.putExtra("DICT", 3);
spec = tabHost.newTabSpec("idzen").setIndicator(tab_menu[2], res.getDrawable(R.drawable.tab_style)).setContent(
tabHost.addTab(spec);

// Select Default Tab
tabHost.setCurrentTab(0);

// Create About Dialog
LayoutInflater inflater = (LayoutInflater) this.getSystemService(LAYOUT_INFLATER_SERVICE);
View layout = inflater.inflate(R.layout.about, (ViewGroup) findViewById(R.id.layout_root));
builder = new AlertDialog.Builder(this)
    .setView(layout)
    .setTitle("About")
    .setNegativeButton("OK", null);
builder.setIcon(android.R.drawable.ic_dialog_info);
aboutDialog = builder.create();

```

Gambar 5. 15 Potongan kode tab menu pedagang atau produk

Selanjutnya untuk melakukan pencarian pedagang membutuhkan *class* cari.java. Untuk kode fungsi pencarian sama dengan method-method yang telah digunakan sebelumnya yaitu menggunakan SQL untuk meng~~excute~~*execute* query pencarian di *database* SQLite. Untuk kode programnya dapat dilihat pada Gambar 5. 16.

```
protected void cari(String text) {
    // TODO Auto-generated method stub
    list_result.clear();
    if (text.length() > 0) {
        String query;
        switch(DICT) {
            default:
                case SEMUA:
                    query = "SELECT Nama_Produk as word FROM Produk where word LIKE '"+text.replace("'", "''"),
                    break;
                case HEWAN:
                    query = "SELECT Nama_Produk FROM Produk where Nama_Produk LIKE '"+text.replace("'", "''"),
                    break;
                case TUMBUHAN:
                    query = "SELECT ID_Pedagang FROM Pedagang where ID_Pedagang LIKE '"+text.replace("'", "''"),
                    break;
        }
        kamusCursor = db.rawQuery(query, null);
        if (kamusCursor.moveToFirst()) {
            list_result.add(kamusCursor.getString(0));
            while(!kamusCursor.isLast()) {
                kamusCursor.moveToNext();
                list_result.add(kamusCursor.getString(0));
            }
        } else {
            list_result.add(not_found);
        }
    }
}

protected void translate(String text) {
```

Gambar 5. 16 Potongan kode pencarian pedagang atau produk.

Untuk mendapatkan informasi *detail* pedagang masih membutuhkan *class* lagi yakni Arti.java. Untuk penjelasan kode program di dalamnya telah dibahas dalam beberapa *class* sebelumnya. Untuk potongan kode programnya dapat dilihat pada Gambar 5. 17.


```

private boolean checkResult() {
    // TODO Auto-generated method stub
    String kingdom1;
    switch(DICT) {
        default:
            case SEMBUA:
                kingdom1 = "SELECT * FROM MainMenu where lower(Type)=lower('"+text.replace("'", "'')+"') ";
                break;
            case HEWAN:
                kingdom1 = "SELECT * FROM Produk where lower>Nama_Produk)=lower('"+text.replace("'", "'')+"')";
                break;
            case TUMBUHAN:
                kingdom1 = "SELECT * FROM Pedagang where lower(ID_Pedagang)=lower('"+text.replace("'", "'')+"')";
                break;
    }
    kingdom = db.rawQuery(kingdom1, null);
    if (!kingdom.moveToFirst()) {
        Toast.makeText(this, "Word '"+text+"' not found in "+dict_name[DICT-1]+" dictionary!", Toast.LENGTH_SHORT);
        return false;
    } else return true;
}

```

Gambar 5. 17 Potongan kode detail pedagang atau produk

5.3.4. Melihat Titik Lokasi Pedagang

Sistem menyediakan fitur untuk melihat titik lokasi pedagang yang diinginkan. Fitur ini diharapkan pengguna dapat menemukan lokasi pedagang dengan mudah. Fitur ini membutuhkan *class* listpeta.java yang didalamnya ada fungsi untuk memanggil (*intent*) *class* lain sesuai daftar *listmenu* (string) yang dipilih. Untuk kode programnya dapat dilihat pada

```

private void tampilkanpilihan(String pilihan) {
    // TODO Auto-generated method stub
    try
    {
        Intent i = null;
        if(pilihan.equals("Pasar Pabean Surabaya"))
        {
            i = new Intent (this,PsPabean.class);
        }
        if(pilihan.equals("Denah Pasar Pabean"))
        {
            i = new Intent (this,DenahPabean.class);
        }

        if(pilihan.equals("Pedagang Ikan Tutik(IK01A)"))
        {
            i = new Intent (this,LokasiPedagangTutik.class);
        }

        startActivity(i);
    }catch(Exception e)
    {
        e.printStackTrace();
    }
}

```

Gambar 5. 18 Potongan kode menampilkan daftar peta.

Class listpeta.java selanjutnya akan memanggil *class* sesuai pilihan user, disini kita ambil contoh pengguna memilih “Pedagang Ikan Tutik(IK01A)”. Untuk kode programnya dapat dilihat pada

```
@Override
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.Lokasipedagangtutik);

    imageView1 = (ImageView)findViewById(R.id.imageView1);
    imageView1.setOnTouchListener(new OnTouchListener() {
```

Gambar 5. 19 Potongan kode menampilkan isi peta yang dipilih (Contoh pedagang Ikan).

5.3.5. Memberikan Informasi Etika di Pasar

Aplikasi *cultural experience guide* menyediakan informasi terkait etika ketika seseorang berada di Pasar Tradisional. Hal ini diharapkan wisatawan yang mengunjungi Pasar Pabean Surabaya memahami budaya yang ada didalamnya terutama terkait dengan cara bertransaksi dengan pedagang. Untuk kode programnya dapat dilihat pada

```
String query = "select * from etika";

db = db_kamus_helper.getReadableDatabase();
cursor = db.rawQuery(query, null);

if(cursor.moveToFirst()){
    do{
        Soal s = new Soal();
        txtis1.setText(cursor.getString(3));
        txtis2.setText(cursor.getString(4));
        txtis3.setText(cursor.getString(5));
        txtis4.setText(cursor.getString(6));
        txtis5.setText(cursor.getString(7));

        jawab.setText(cursor.getString(8));
        jawab2.setText(cursor.getString(9));
        jawab3.setText(cursor.getString(10));
        jawab4.setText(cursor.getString(11));
        jawab5.setText(cursor.getString(12));

        judul.setText(cursor.getString(2));
        listSoal.add(s);
    }while(cursor.moveToNext());
}

return listSoal;
}
```

Gambar 5. 20 Potongan kode EtikaPasar.java.

5.3.6. Memberikan Informasi Dialek Jawa di Pasar

Dialek Jawa merupakan fitur penting di aplikasi *cultural experience guide* karena tiap daerah memiliki bahasa lokal yang berbeda-beda. Dengan adanya fitur dialek jawa mempermudah wisatawan dalam mempelajari kebudayaan setempat terutama dalam berkomunikasi dengan pedagang yang ada di Pasar Pabean Surabaya. Transaksi akan jauh lebih mudah dengan adanya fitur dialek jawa yang memberikan informasi kumpulan dialek Jawa Surabaya, Transaksi tawar-menawar, dan Angka Jawa. Tidak hanya berbentuk sebuah informasi namun dialek-dialek yang ada bisa didengarkan sesuai logat (pengucapan asli) Jawa Surabaya. Untuk menampilkan informasi kumpulan Dialek Jawa Surabaya, untuk kode programnya dapat dilihat pada Gambar 5. 21.

```
@Override
public void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.kamusjawa);
    db_kamus_helper = new DBKamus(this);
    try {
        db_kamus_helper.openDataBase();
        db = db_kamus_helper.getReadableDatabase();
    } catch (SQLException sqle){
        throw sqle;
    }
}
```

Gambar 5. 21 Method untuk megambil data di *class* KamusDialek

Untuk menampilkan isi Dialek Jawa dari *database* membutuhkan query SQL. Method yang digunakan sama dengan method yang digunakan di *class* lain yang telah dibahas sebelumnya. Untuk kode programnya dapat dilihat pada Gambar 5. 22.

```

List<Soal> listSoal = new ArrayList<Soal>();
String query = "select * from kamusjawa";

db = db_kamus_helper.getReadableDatabase();
cursor = db.rawQuery(query, null);

if(cursor.moveToFirst()){
    do{
        Soal s = new Soal();
        txtisi.setText(cursor.getString(2));

        judul.setText(cursor.getString(1));|
        listSoal.add(s);
    }while(cursor.moveToNext());
}

return listSoal;
}

```

Gambar 5. 22 Potongan kode menampilkan kumpulan Dialek Jawa.

5.3.7. Memberikan Informasi Dialek Pasar Antara Pedagang dan Pembeli

Informasi dialek antara pedagang dan pembeli merupakan fitur utama yang harus ada di aplikasi *cultural experience guide*. Fitur ini memberikan pengalaman budaya setempat kepada wisatawan /pengguna aplikasi dengan adanya percakapan atau dialek Jawa Surabaya yang diambil sesuai budaya setempat yang bisa didengarkan. Dengan demikian wisatawan akan jauh lebih memahami dialek-dialek yang didengar saat berkunjung ke Pasar Pabean Surabaya. Diharapkan bahasa/dialek Jawa bukan lagi menjadi penghalang untuk melakukan transaksi antara pedagang dan pembeli. Secara garis besar Dialek Jawa Pedagang dan Pembeli dibagi kedalam tiga jenis, yaitu Dialek Jawa Pedagang Ikan, Dialek Jawa Pedagang Buah, dan Dialek Jawa Pedagang Sayur. Untuk kode program menampilkan Dialek Jawa Pedagang Ikan dapat dilihat pada Gambar 5. 23.

```

public List<Soal> getSoal(){
    List<Soal> listSoal = new ArrayList<Soal>();
    String query = "select * from dialekikan";

    db = db_kamus_helper.getReadableDatabase();
    cursor = db.rawQuery(query, null);

    private void loadClip() {
        try {
            mp=MediaPlayer.create(this, R.raw.dialek);
            mp.setOnCompletionListener(this);

            media=MediaPlayer.create(this, R.raw.dialekduapuluh);
            media.setOnCompletionListener(this);

            media3=MediaPlayer.create(this, R.raw.dialekenambelas);
            media3.setOnCompletionListener(this);

            media4=MediaPlayer.create(this, R.raw.dialekempatbelas);
            media4.setOnCompletionListener(this);

            media5=MediaPlayer.create(this, R.raw.dialekduasembilan);
            media5.setOnCompletionListener(this);

```

Gambar 5. 23 Potongan kode menampilkan dialek Jawa Pedagang Ikan.

Untuk menampilkan Dialek Jawa Pedagang buah membutuhkan *class* DialekBuah.java. Untuk kode programnya dapat dilihat pada Gambar 5. 24.

```

public List<Soal> getSoal(){
    List<Soal> listSoal = new ArrayList<Soal>();
    String query = "select * from dialekbuah";

    db = db_kamus_helper.getReadableDatabase();
    cursor = db.rawQuery(query, null);

    private void loadClip() {
        try {
            mp=MediaPlayer.create(this, R.raw.papatsiji);
            mp.setOnCompletionListener(this);

            media=MediaPlayer.create(this, R.raw.pitulas);
            media.setOnCompletionListener(this);

```

Gambar 5. 24 Potongan kode menampilkan dialek Jawa pedagang buah.

Untuk menampilkan Dialek Jawa Pedagang sayur membutuhkan *class* DialekSayur.java. Untuk kode programnya dapat dilihat pada

```
public List<Soal> getSoal(){
    List<Soal> listSoal = new ArrayList<Soal>();
    String query = "select * from dialeksayur";

    db = db_kamus_helper.getReadableDatabase();
    cursor = db.rawQuery(query, null);

private void loadClip() {
    try {
        mp=MediaPlayer.create(this, R.raw.tuwek);
        mp.setOnCompletionListener(this);

        media=MediaPlayer.create(this, R.raw.enom);
        media.setOnCompletionListener(this);

        media3=MediaPlayer.create(this, R.raw.piroiki);
        media3.setOnCompletionListener(this);
```

Gambar 5. 25 Potongan kode menampilkan dialek Jawa pedagang sayur

5.3.8. Melihat dan Menentukan Informasi Petunjuk Arah ke Pasar Pabean Surabaya.

Aplikasi *cultural experience guide* dilengkapi dengan fitur navigasi untuk memudahkan wisatawan/pengguna dalam mencari rute perjalanan dengan tujuan Pasar Pabean Surabaya. Wisatawan cukup menginputkan lokasi titik awal ke dalam aplikasi, secara otomatis sistem akan menampilkan rute perjalanan dari titik awal ke titik lokasi yang dituju. Fitur ini memanfaatkan Google Maps API sehingga lokasi peta yang ditampilkan oleh sistem sesuai dengan kondisi nyata. Untuk menampilkan halaman navigasi rute perjalanan Pasar Pabean Surabaya membutuhkan *class* PspPabean.java. Untuk kode programnya dapat dilihat pada

```

public class PsPabean extends Activity {

    @Override
    protected void onCreate(Bundle
savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.pspabean);

        WebView browser = (WebView)
findViewById (R.id.pabean);

        browser.getSettings().setJavaScriptEnabled(true);

        browser.loadUrl("file:///android_asset/RutePer
jalanan.html");
    }
}

```

Gambar 5. 26 Potongan kode memanggil halaman HTML.

Untuk memanfaatkan Google Maps API, aplikasi *cultural experience guide* menggunakan *toolkits* GUI yang dimiliki oleh librari android. Class PsPabean.java akan menampilkan halaman HTML di widget *WebView*. Halaman HTML di *Project Android* nantinya akan disimpan di folder *asset*. Untuk detail kode program dari halaman RutePerjalanan.html dapat dilihat pada Gambar 5. 27.

```

var directionsDisplay;
var directionsService = new google.maps.DirectionsService();
var map;

function initialize() {
    directionsDisplay = new google.maps.DirectionsRenderer();
    var myLatLng = new google.maps.LatLng(-7.235263, 112.741828);
    var mapOptions = {
        zoom: 16,
        center: myLatLng
    }
    map = new google.maps.Map(document.getElementById('map-canvas'), mapOptions);
    directionsDisplay.setMap(map);
    var marker = new google.maps.Marker({
        position: myLatLng,
        map: map,
        title: 'Hello World!'
    });
}

```

Gambar 5. 27 Potongan kode halaman HTML untuk menggunakan Goole Maps API.

Sedangkan kode program untuk membuat petunjuk arah ke Pasar Pabean Surabaya memanfaatkan Google API dapat dilihat pada Gambar 5. 28

```
function calcRoute() {
    var start = document.getElementById('start').value;
    var end = document.getElementById('end').value;
    var request = {
        origin:start,
        destination:end,
        travelMode: google.maps.TravelMode.DRIVING
    };
    directionsService.route(request, function(response, status) {
        if (status == google.maps.DirectionsStatus.OK) {
            directionsDisplay.setDirections(response);
        }
    });
}

google.maps.event.addDomListener(window, 'load', initialize);
```

Gambar 5. 28 Potongan kode program untuk petunjuk arah ke Pasar Pabean Surabaya

5.4. Uji Coba

Bagian ini menjelaskan tentang uji coba yang akan dilakukan pada aplikasi ini. Uji coba yang dilakukan adalah Uji Coba Fungsional, Uji Coba Performa, dan Uji Coba Penerimaan Pengguna.

5.4.1. Uji Coba Fungsional

Uji coba fungsional bertujuan untuk mengetahui apakah fungsi-fungsi yang dibuat sudah berjalan sesuai rancangan aplikasi. Uji coba fungsional dilakukan sesuai dengan format test case yang bisa dilihat pada Tabel 5. 3 . Adapun daftar testcase dapat dilihat pada Tabel 5. 4.

Tabel 5. 3 Format Test Case

No	Aksi Test	Data Test	Hasil yang Diharapkan	Hasil Aktual	Sukses / Gagal
1	User input nama produk yang diinginkan	Cari nama = nama produk	User masuk halaman detail produk yang diinput.	Sistem menam pilkan halaman detail produk	Sukses

Tabel 5. 4 Daftar Test Case

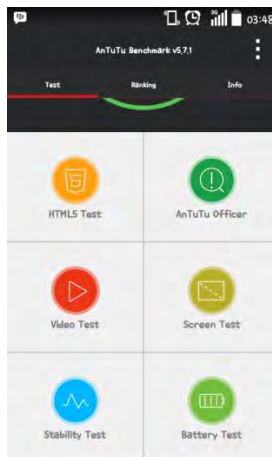
Kode Test Case	Nama Test Case
TC-01	Load halaman utama aplikasi
TC-02	Melihat daftar produk
TC-03	Melihat daftar pedagang
TC-04	Mencari nama produk
TC-05	Mencari nama pedagang
TC-06	Melihat dan mendengarkan Kumpulan Dialek Jawa
TC-07	Melihat dan mendengarkan dialek Jawa pedagang ikan
TC-08	Melihat dan mendengarkan dialek Jawa pedagang buah
TC-09	Melihat dan mendengarkan dialek Jawa pedagang sayur
TC-10	Melihat peta lokasi Pasar Pabean Surabaya dan mencari rute perjalanan ke Pasar Pabean Surabaya
TC-11	Melihat denah Pasar Pabean Surabaya
TC-12	Melihat lokasi pedagang
TC-13	Melihat informasi etika di pasar
TC-14	Melihat informasi terkait kriminalitas di Pasar Pabean Surabaya.
TC-15	Melihat informasi Suku dan Kebiasaan pedagang di Pasar Pabean Surabaya.
TC-16	Melihat dan mencari harga komoditi produk yang ada di Pasar Pabean Surabaya.
TC-17	Melihat dan mendengarkan angka Jawa Surabaya

5.4.2. Uji Coba Performa

Uji coba performa dilakukan untuk mengetahui kemampuan aplikasi dalam mengolah permintaan dari user. Aplikasi *cultural experience guide* menggunakan *platform mobile*. Beberapa pengujian dilakukan diantaranya mencari dan

menampilkan detail produk, mencari informasi harga komoditi, serta mencari lokasi rute perjalanan dan menampilkan lokasi Pasar Pabean Surabaya. Pengujian dilakukan untuk mengukur seberapa cepat sistem menampilkan permintaan user atau klien. Sistem menggunakan database SQLite yang telah disediakan oleh sistem operasi android. Kecepatan untuk menampilkan permintaan user tergantung dari versi android dan spesifikasi deviceny. Untuk menguji performa android bisa menggunakan AnTuTuBenchmark [21].

Pengujian *performance* device dengan aplikasi AnTuTu Benchmark dapat dilihat pada Gambar 5. 29. Untuk mengukur performa hasil permintaan dari user yang membutuhkan koneksi internet. Pengujian pencarian informasi harga komoditi, serta mencari lokasi rute perjalanan dan menampilkan lokasi Pasar Pabean dapat dilakukan dengan menggunakan aplikasi Internet Speed Meter. [22]



Gambar 5. 29 Aplikasi Uji Tes Benchmark AnTuTu Benchmark.

5.4.3. Uji Coba Penerimaan pengguna

Uji coba penerimaan pengguna digunakan untuk mengetahui *feedback* dari calon pengguna aplikasi sebelum diluncurkan untuk umum. Uji coba penerimaan pengguna dilakukan berdasarkan format dokumen *User Acceptance Testing* (UAT) dari *Swiburne University of Technology*. Dokumen ini berisikan *checklist* dari tiap-tiap fungsi yang ada dalam aplikasi. Pengujian dilakukan pada 7 orang responden. Responden akan dipandu untuk mengerjakan *task* yang ada dalam dokumen, lalu memberikan penilaian apakah fitur tersebut sudah sesuai (*pass*) atau masih perlu diperbaiki (*fail*). Apabila masih perlu perbaikan, maka responden diharuskan mengisi alasan kenapa fitur tersebut masih perlu perbaikan. Bentuk format dokumen UAT yang digunakan dapat dilihat pada lampiran F.

(halaman ini sengaja dikosongkan)

BAB VI

HASIL DAN PEMBAHASAN

6.1. Uji Coba

Pada bagian ini akan dijelaskan hasil uji coba yang telah dilakukan terhadap aplikasi. Hasil uji coba meliputi hasil uji coba fungsional dari aplikasi dengan menggunakan *test case* dan hasil uji coba performa dengan menggunakan aplikasi AnTutu Benchmark

6.1.1. Uji Coba Fungsional

Secara teknis pengujian menggunakan *test case*. Untuk *test case* sudah dijelaskan di bab sebelumnya. Hasil uji coba dapat dilihat pada Tabel 6. 1 . Untuk *detail* dari setiap *test case* yang ada bisa dilihat pada lampiran G.

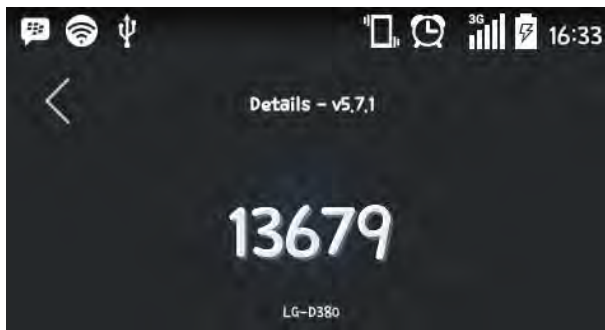
Tabel 6. 1 Hasil Uji Coba Fungsional

Kode Test Case	Nama Test Case	Status
TC-01	Load halaman utama aplikasi	Terpenuhi
TC-02	Melihat daftar produk	Terpenuhi
TC-03	Melihat daftar pedagang	Terpenuhi
TC-04	Mencari nama produk	Terpenuhi
TC-05	Mencari nama pedagang	Terpenuhi
TC-06	Melihat dan mendengarkan Kumpulan Dialek Jawa	Terpenuhi
TC-07	Melihat dan mendengarkan dialek Jawa pedagang ikan	Terpenuhi
TC-08	Melihat dan mendengarkan dialek Jawa pedagang buah	Terpenuhi
TC-09	Melihat dan mendengarkan dialek Jawa pedagang sayur	Terpenuhi
TC-10	Melihat peta lokasi Pasar Pabean Surabaya dan mencari rute perjalanan ke Pasar Pabean Surabaya	Terpenuhi

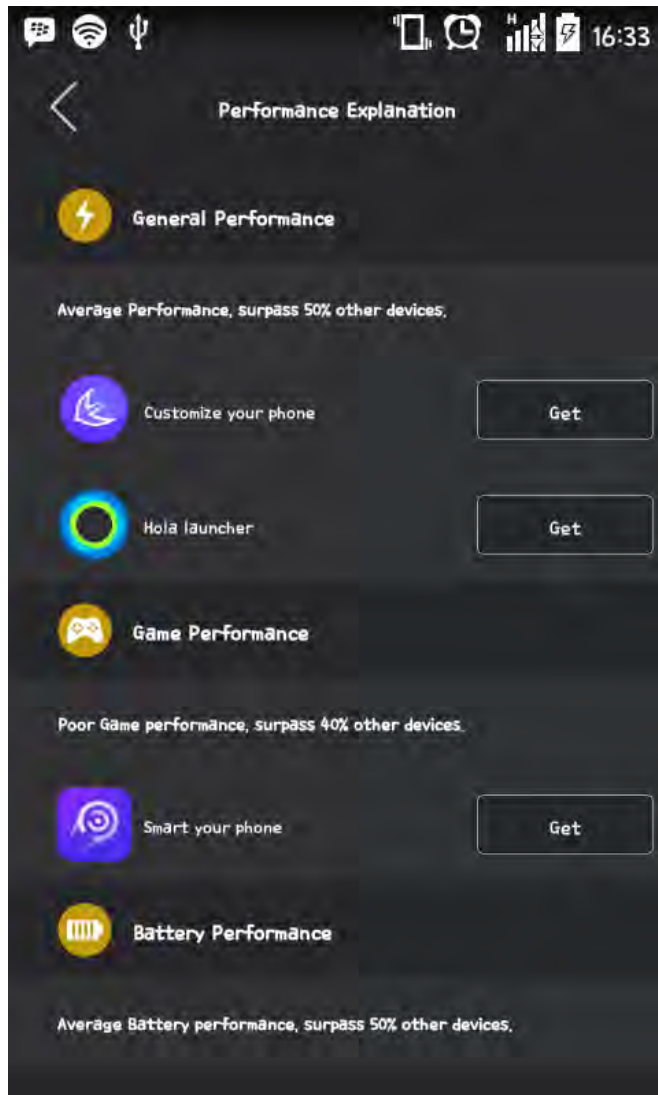
Kode Test Case	Nama Test Case	Status
TC-11	Melihat denah Pasar Pabean Surabaya	Terpenuhi
TC-12	Melihat lokasi pedagang	Terpenuhi
TC-13	Melihat informasi etika di pasar	Terpenuhi
TC-14	Melihat informasi terkait kriminalitas di Pasar Pabean Surabaya.	Belum Terpenuhi
TC-15	Melihat informasi Suku dan Kebiasaan pedagang di Pasar Pabean Surabaya.	Terpenuhi
TC-16	Melihat dan mencari harga komoditi produk yang ada di Pasar Pabean Surabaya.	Terpenuhi
TC-17	Melihat dan mendengarkan angka Jawa Surabaya	Terpenuhi

6.1.2. Hasil Uji Coba Performa

Uji coba performa aplikasi *cultural experience guide* dilakukan melalui device yang digunakan karena database yang digunakan SQLite maka untuk mengukur kecepatan untuk menampilkan permintaan user tergantung dari versi android dan spesifikasi devicenya. Hasil kesimpulan uji coba dari device yang digunakan memberikan nilai untuk tes performa dapat dilihat pada Gambar 6. 1.



Gambar 6. 1 Hasil Tes Uji Performa Device



Gambar 6. 2 *Detail Hasil Uji Performa Device*



Gambar 6. 3 Info *Detail* Pengujian Device

6.1.3. Hasil Uji Coba Penerimaan Pengguna

Uji coba penerimaan pengguna dilakukan berdasarkan format *User Acceptance Testing* (UAT) dari *Swburne University of Technology*. Pengujian dilakukan pada 7 orang responden. Uji coba dilakukan pada 3 bagian yakni uji coba fitur yang ada dalam aplikasi (Tabel 6. 2), hasil uji coba tiap responden (Tabel 6. 3) dan *feedback* dari responden secara garis besar(Tabel 6. 4).

Tabel 6. 2 Hasil Uji Coba Tiap Fitur

No	Fitur	Puas	Tidak Puas
Halaman Utama			
1	Splashscreen menu	7	0
2	Menampilkan menu utama	5	2
Menu Utama			
3	Menampilkan daftar produk	5	2
4	Menampilkan daftar pedagang	5	2
5	Menampilkan dan mendengarkan dialek Jawa Surabaya	5	2
6	Menampilkan denah Pasar Pabean Surabaya	3	4
7	Menampilkan rute perjalanan menuju Pasar Pabean Surabaya.	7	0
8	Mencari informasi tiap pedagang secara rinci	6	1
9	Mencari informasi tiap produk secara rinci.	6	1
10	Melihat dan mencari informasi harga komoditi terkini	7	0
11	Menampilkan denah pedagang	3	4
Sub Menu atau Lain-lain			
12	Melihat etika di pasar	6	1

No	Fitur	Puas	Tidak Puas
13	Melihat suku dan kebiasaan pedagang (secara garis besar)	4	3
14	Melihat informasi kriminalitas	5	2
Total		74	24

Tabel 6. 3 Hasil Uji Coba untuk Responden

No	Nama Responden	Puas	Tidak Puas
1	Dian Roseno	9	5
2	Sumirah	12	2
3	Titiek	11	3
4	Setya Ningsih	10	4
5	Dwi Hariyanti	11	3
6	Kholifah	10	4
7	Puji Astuti	11	3
Total		74	24

Tabel 6. 4 Hasil *feedback* pengguna

No	Komentar / <i>feedback</i>
#	Halaman Utama
1	Splash Screen untuk halaman utama sudah menarik, disertai proses <i>load database</i> aplikasi.
2	Menu halaman utama bisa dibuat lebih menarik tidak hanya button, untuk Tab menu sudah cukup menarik
#	Menu Utama
1	Untuk daftar produk dan pedagang tampilan <i>background</i> bisa diganti dengan warna yang lebih cerah.
2	Tampilan denah Pasar Pabean Surabaya bisa dibuat peta 3D dan lebih akurat, sehingga akan tampak sangat menarik
3	Semua informasi terkait pedagang dan produk kurang banyak.

No	Komentar / <i>feedback</i>
4	Audio dialek Jawa tidak animasi orang melakukan percakapan, jika ada akan sangat lebih menarik
	Denah lokasi pedagang dibuat peta 3D dan ada <i>marker</i> yang menyala untuk lokasi pedagang yang diinginkan
#	<i>Sub Menu</i> atau Lain-lain
1	Informasi etika di pasar, suku dan kebiasaan pedagang masih minim
2	Informasi kriminalitas kurang banyak

6.2. Pembahasan Uji Coba

Pada bagian ini akan dijelaskan analisa pembahasan dari hasil uji coba yang telah dijabarkan di sub-bab sebelumnya. Pembahasan uji coba meliputi pembahasan hasil uji coba fungsional dan pembahasan hasil uji coba performa.

6.2.1. Pembahasan Hasil Uji Coba Fungsional

Berdasarkan hasil uji coba, dapat dilihat bahwa hampir semua fitur *test case* dari aplikasi terpenuhi. Namun masih terdapat fitur yang masih belum sepenuhnya terpenuhi, yaitu pada fitur melihat informasi terkait kriminalitas di Pasar Pabean Surabaya (TC-14). Pada fitur ini untuk menampilkan informasi kurang lengkap dikarenakan membutuhkan sumber informasi yang banyak terkait kriminalitas yang ada di Pasar Pabean Surabaya masih kurang.

Dari hasil uji coba yang telah dilakukan, aplikasi *cultural experience guide* sudah mampu memenuhi hampir semua fitur pada *test case* kecuali untuk pada fitur TC-14, kekurangannya terdapat pada sumber informasi. Namun secara keseluruhan aplikasi ini bisa dikatakan sudah cukup mampu berjalan dengan baik.

6.2.2. Pembahasan Hasil Uji Coba Performa

Berdasarkan hasil uji coba peforma yang telah dilakukan, dapat dilihat bahwa sistem mendapat request mencari dan menampilkan detail produk, mendengarkan dialek jawa, mencari informasi harga komoditi, serta mencari lokasi rute perjalanan dan menampilkan lokasi Pasar Pabean Surabaya tergantung dari device yang digunakan karena menggunakan database SQLite.

Namun ada beberapa pengecualian untuk request, yakni mencari informasi harga komoditi, serta mencari lokasi rute perjalanan dan menampilkan lokasi Pasar Pabean Surabaya membutuhkan koneksi internet untuk mendapatkan data. Kecepatan sistem untuk mengolah request pastinya akan tergantung kecepatan internet dari setiap *provider* yang digunakan.

6.2.3. Pembahasan Hasil Uji Coba Penerimaan Pengguna

Berdasarkan hasil uji coba penerimaan pengguna, dapat dilihat bahwa terdapat 74 jawaban puas dibanding 24 jawaban tidak puas dari 7 responden yang terbagi pada 14 fitur yang diujikan. Pada tabel hasil uji coba per fitur, dapat dilihat bahwa ada 3 fitur yang semua responden menjawab puas yaitu pada fitur melihat *Splashscreen menu*, Menampilkan rute perjalanan menuju Pasar Pabean Surabaya. dan Melihat dan mencari informasi harga komoditi terkini.

Berdasarkan tabel hasil *feedback* pengguna, maka dapat dilakukan beberapa perbaikan untuk penelitian selanjutnya seusai dengan yang tertera pada tabel. Berdasarkan hasil uji coba penerimaan pengguna secara keseluruhan maka bisa dikatakan bahwa aplikasi ini sudah mampu diterima dengan baik dengan melihat jumlah kepuasan yang lebih banyak dibandingkan ketidakpuasan, namun masih perlu ditekankan fitur yang masih menerima respon tidak puas yang lebih

banyak serta beberapa fitur yang masih belum memuaskan pengguna secara keseluruhan. Dikarenakan kurangnya waktu penelitian, maka diharapkan *feedback* yang didapatkan dari uji coba ini dapat dijadikan masukan untuk penelitian selanjutnya.

(halaman ini sengaja dikosongkan)

BAB VII

KESIMPULAN DAN SARAN

7.1. Kesimpulan

Dari hasil pengembangan aplikasi *Cultural Experience Guide* berbasis Android dapat diambil beberapa kesimpulan, yaitu:

- 1) Aplikasi *Cultural Experience Guide* dapat digunakan untuk membantu para wisatawan sebagai pemandu pengalaman kultural dengan konteks budaya Jawa.
- 2) Aplikasi *Cultural Experience Guide* dilengkapi fitur navigasi dengan menggunakan Google Map, sehingga bisa digunakan sebagai *Tour Guide* digital yang praktis.
- 3) Aplikasi *Cultural Experience Guide* telah diinstal pada beberapa perangkat Android dengan spesifikasi berbeda dan berfungsi sebagaimana fungsinya, namun ketersediaan layanan Internet pada perangkat menjadi faktor yang harus diperhatikan dalam penggunaannya, sehingga aplikasi dapat berjalan dengan baik.
- 4) Pembuatan Aplikasi *Cultural Experience Guide* terbilang tidak terlalu sulit, khususnya pada aplikasi-aplikasi dengan fitur sederhana karena hanya menggunakan method-method dari kelas yang sudah ada yang disediakan Android pada library-library dalam Android SDK.
- 5) Aplikasi *Cultural Experience Guide* dilengkapi dengan fitur audio, dimana pengguna dapat mendengarkan percakapan dialek Jawa yang digunakan pedagang dan pembeli dalam melakukan transaksi jual-beli.

- 6) Aplikasi *Cultural Experience Guide* dilengkapi dengan adanya denah Pasar Pabean Surabaya, sehingga memudahkan pengguna dalam mencari lokasi pedagang yang diinginkan.
- 7) Aplikasi ini menggunakan *webkit*, dimana menampilkan format HTML sebagai *widget* didalam android yang menampilkan informasi harga. Informasi ini sifatnya dinamis karena sinkron dengan website untuk pedagang *update* informasi harga komoditi terkini.

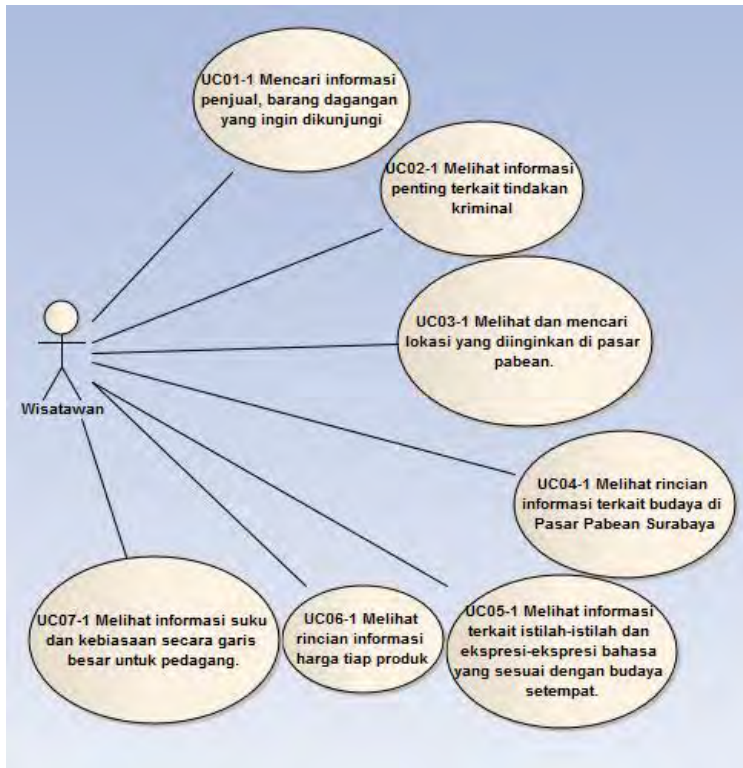
7.2. Saran

Dari peninjauan yang telah dilakukan oleh peneliti masih terdapat beberapa kekurangan pada aplikasi *cultural experience guide* yang perlu diperbaiki dan dikembangkan lagi, maka terdapat beberapa saran yang dapat dijadikan masukan untuk penelitian selanjutnya, yaitu:

- 1) Dilihat dari *feedback* uji coba penerimaan pengguna Saran dan komentar responden dapat dipertimbangkan ke penelitian selanjutnya untuk dilakukan pengembangan dan penyempurnaan terhadap fitur yang sudah ada.
- 2) Sistem ini bisa ditambahkan server eksternal sebagai media penyimpanan *resource*, sehingga tidak memberatkan ukuran aplikasi.

LAMPIRAN A

A.1. Use Case Model untuk Wisatawan



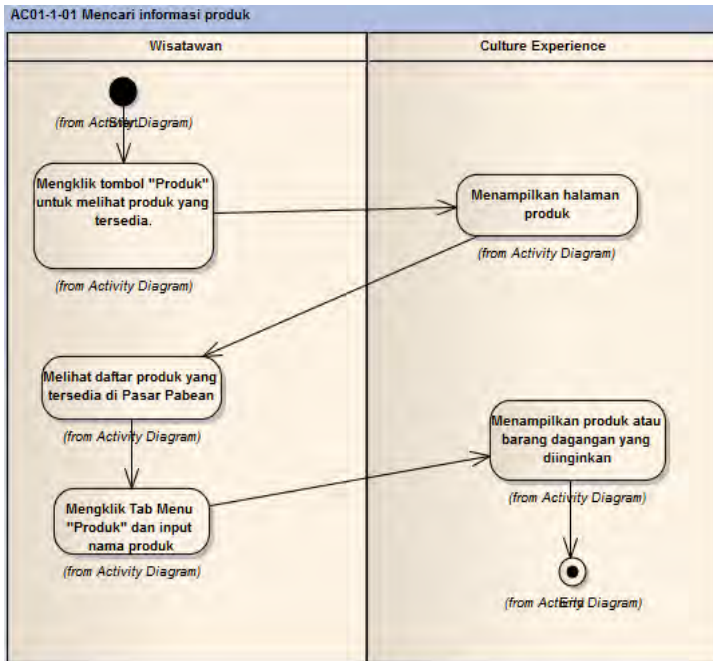
Gambar A. 1 Use Case Model Wisatawan

A2

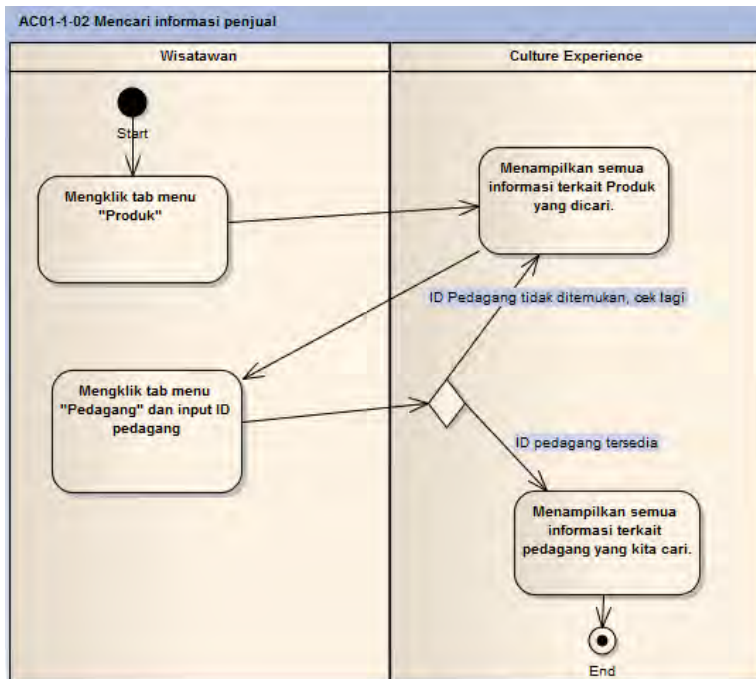
(halaman ini sengaja dikosongkan)

LAMPIRAN B

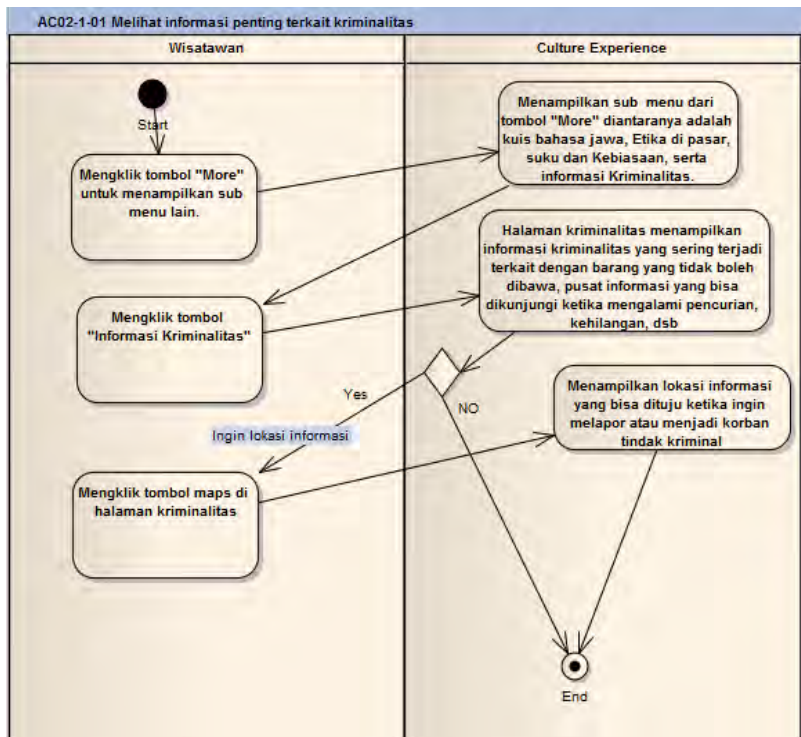
AC01-1-01 Mencari informasi produk



Gambar B. 1 Activity Diagram mencari informasi produk

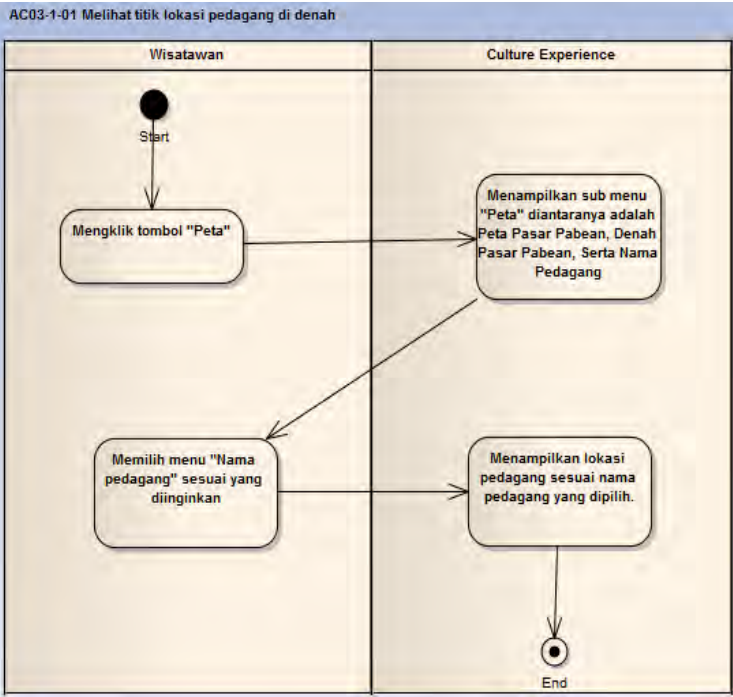
AC01-1-02 Mencari informasi penjual**Gambar B. 2 Activity Diagram mencari Informasi penjual**

AC02-1-01 Melihat informasi penting terkait kriminalitas



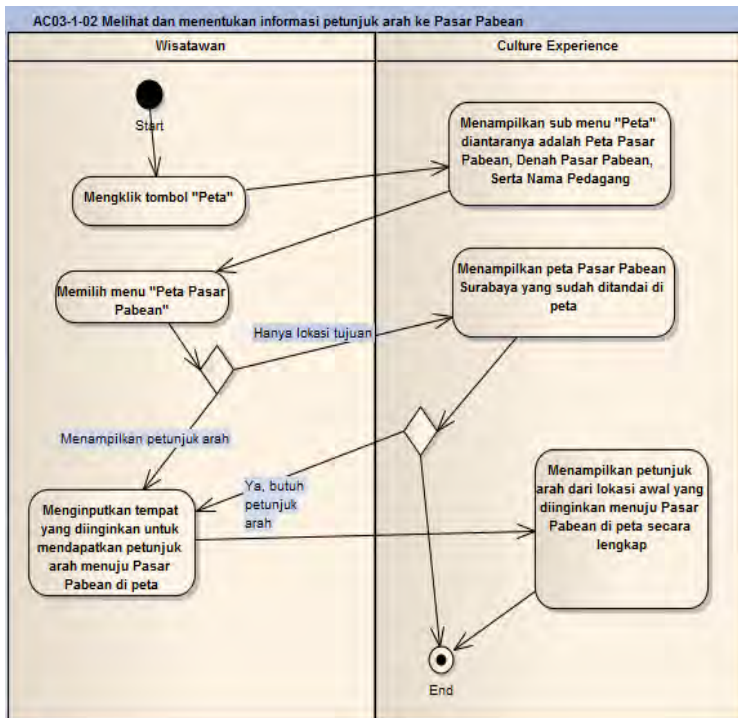
Gambar B. 3 Activity Diagram melihat informasi penting terkait kriminalitas

AC03-1-01 Melihat titik lokasi pedagang di denah



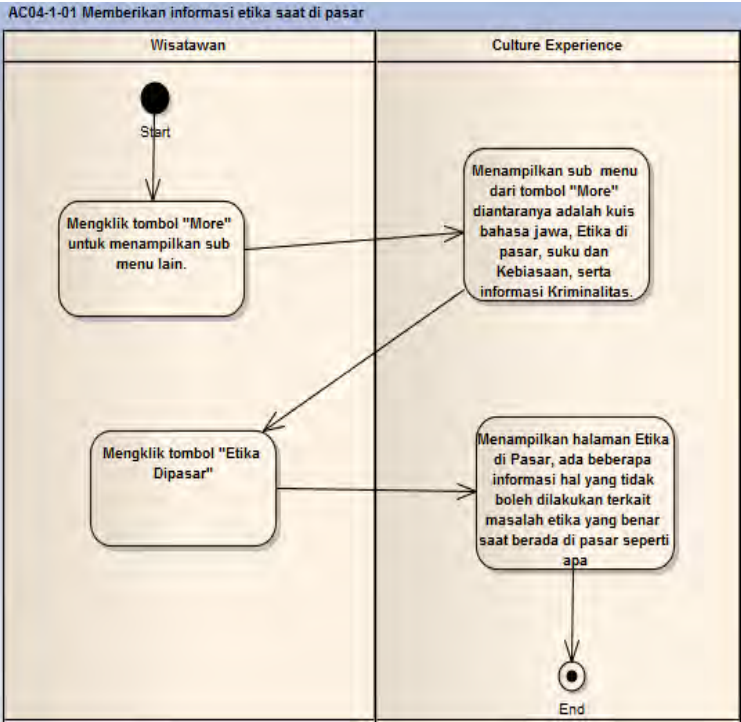
Gambar B. 4 Activity Diagram melihat titik lokasi pedagang di denah

AC03-1-02 Melihat dan menentukan informasi petunjuk arah ke Pasar Pabean



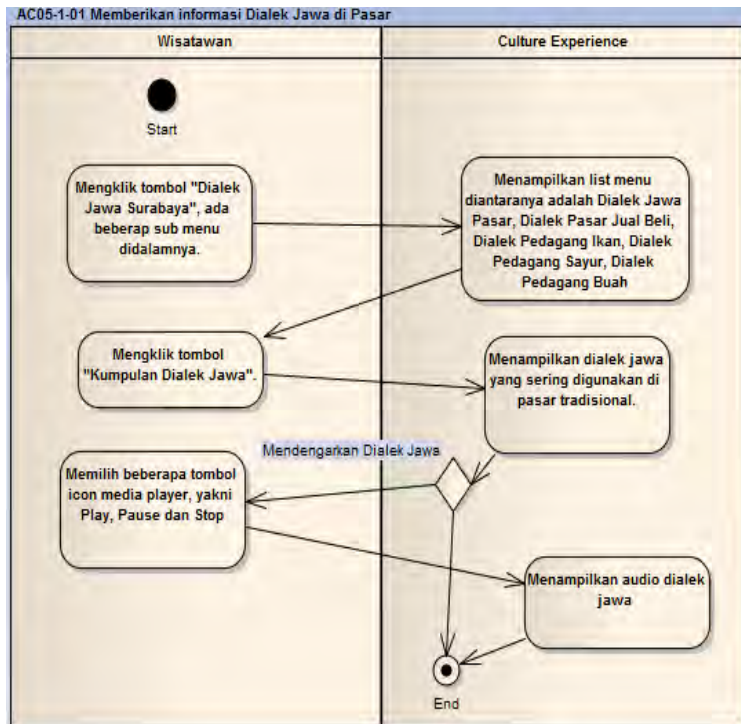
Gambar B. 5 Activity Diagram melihat dan menentukan informasi petunjuk arah ke Pasar Pabean

AC04-1-01 Memberikan informasi etika saat di pasar



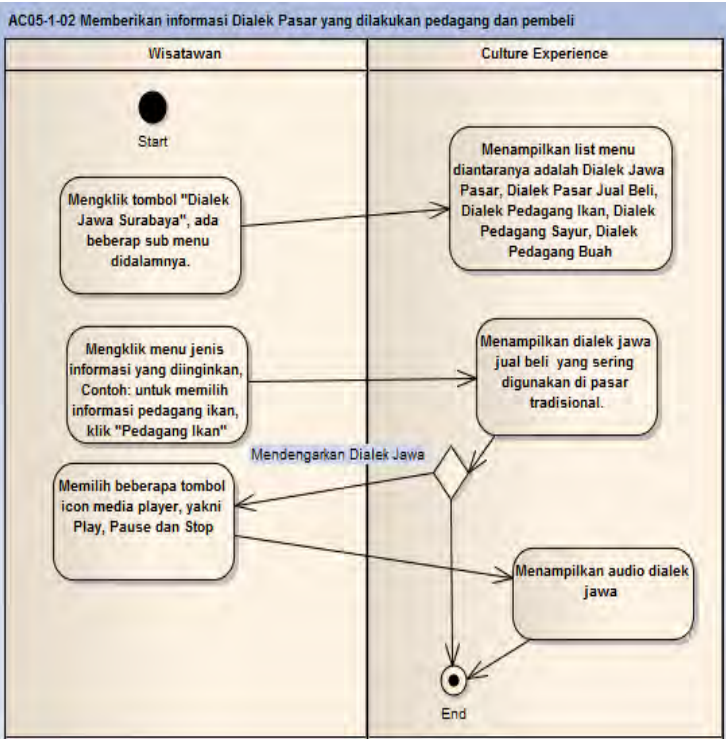
Gambar B. 6 Activity Diagram memberikan informasi etika saat di pasar

AC05-1-01 Memberikan informasi Dialek Jawa di Pasar



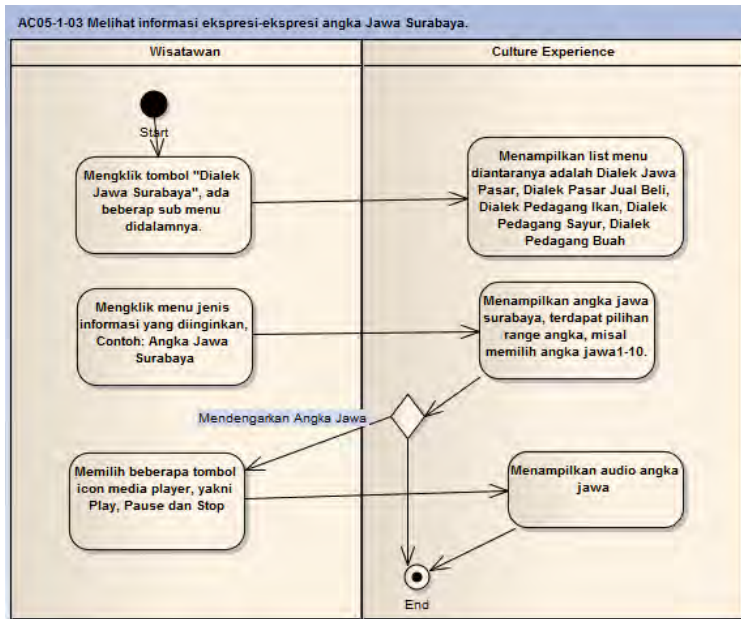
Gambar B. 7 Activity Diagram memberikan informasi Dialek Jawa di Pasar

AC05-1-02 Memberikan informasi Dialek Pasar yang dilakukan pedagang dan pembeli



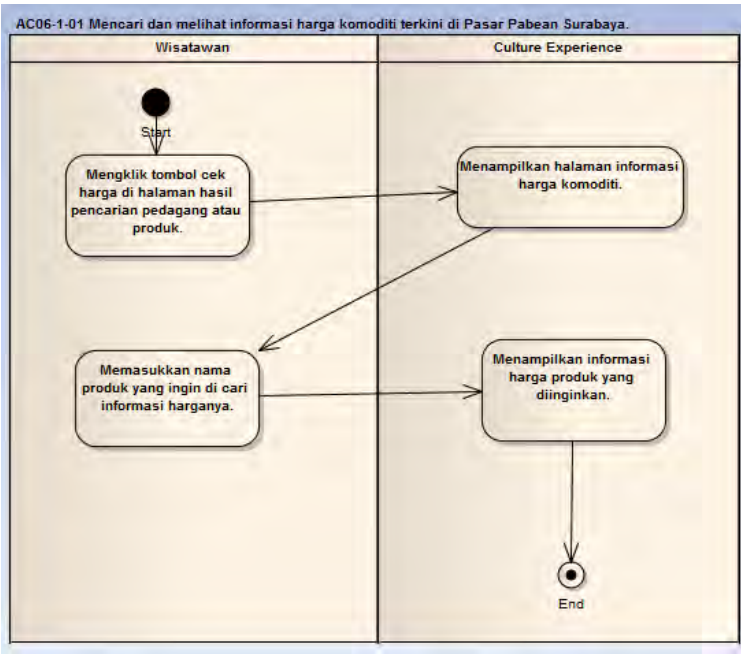
Gambar B. 8 Activity Diagram memberikan informasi Dialek Pasar jual-beli

AC05-1-03 Melihat informasi ekspresi-ekspresi dialek angka Jawa.



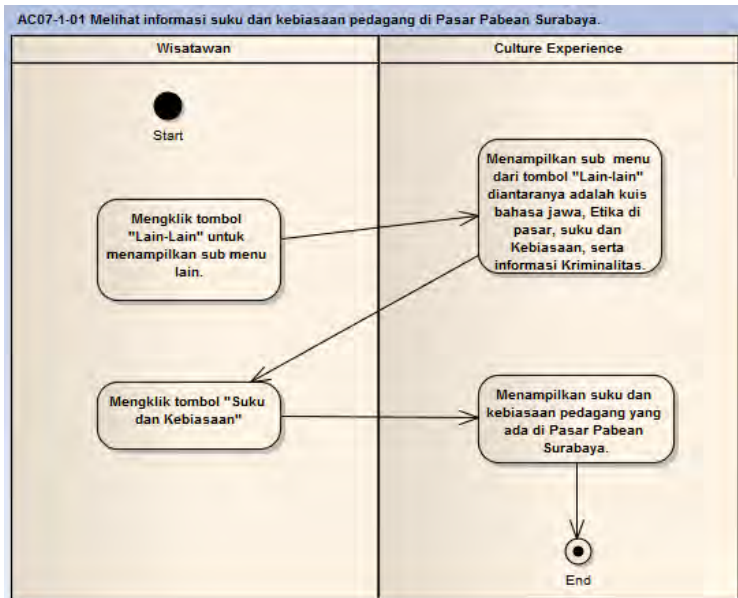
Gambar B. 9 Melihat informasi ekspresi-ekspresi dialek angka Jawa.

AC06-1-01 Mencari dan melihat informasi harga terkini di Pasar Pabean Surabaya.



Gambar B. 10 Mencari dan melihat informasi harga terkini di Pasar Pabean Surabaya.

AC07-1-01 Melihat informasi suku dan kebiasaan pedagang di Pasar Pabean Surabaya.



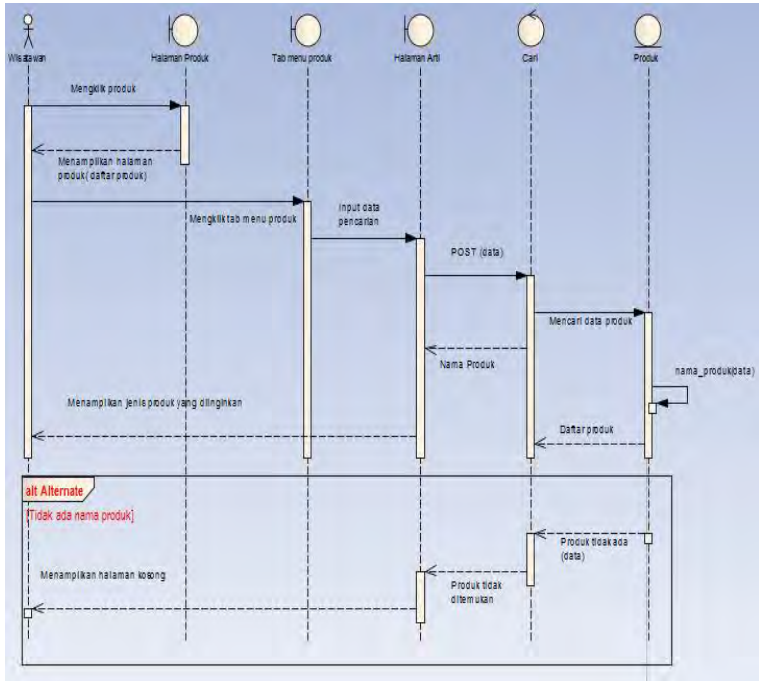
Gambar B. 11 Melihat informasi suku dan kebiasaan pedagang di Pasar Pabean Surabaya

B12

(halaman ini sengaja dikosongkan)

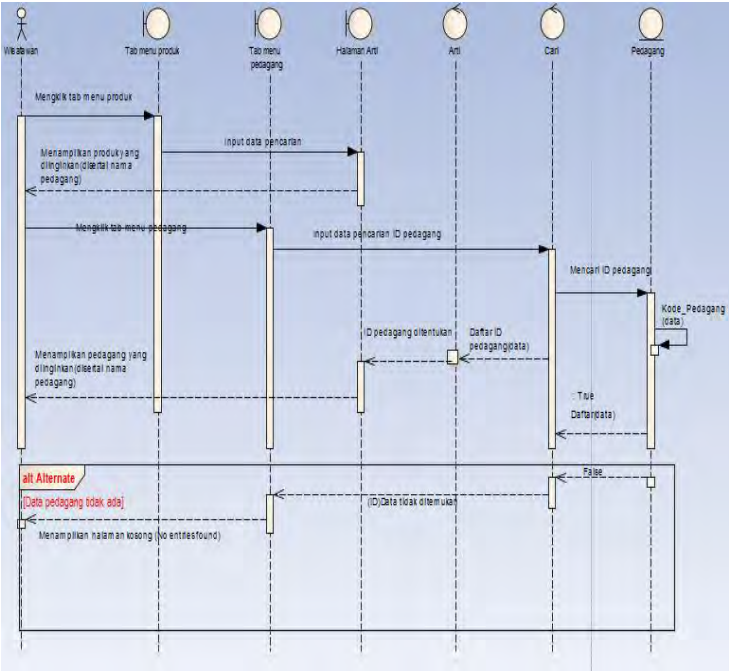
LAMPIRAN C

SD01-1-01 Mencari informasi produk



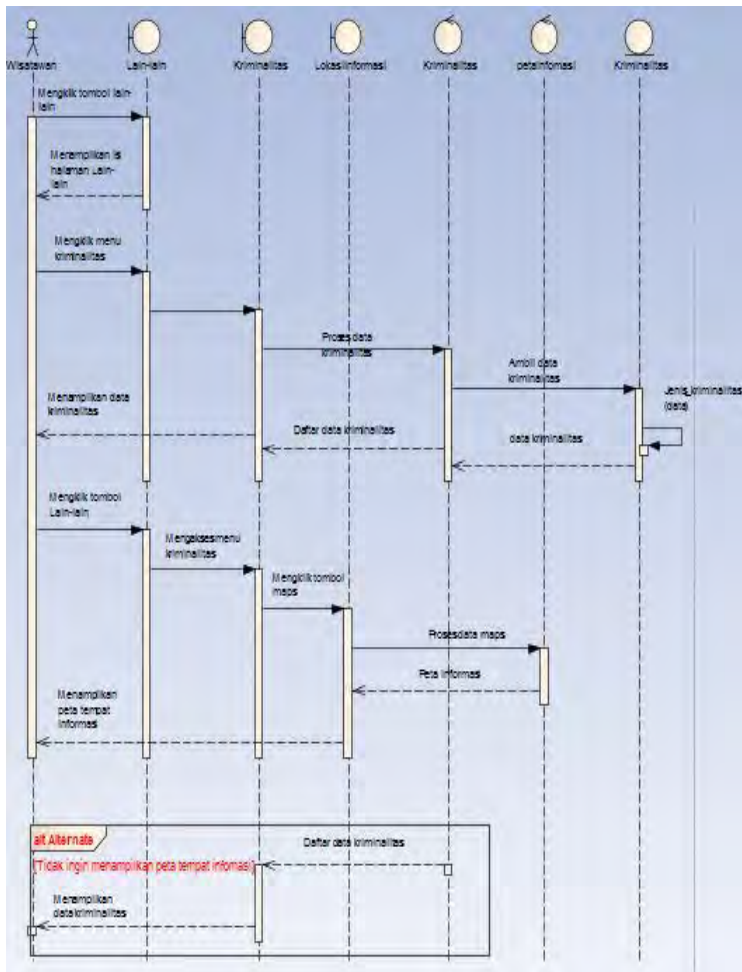
Gambar C. 1 Sequence Diagram mencari informasi produk.

SD01-1-02 Mencari informasi penjual

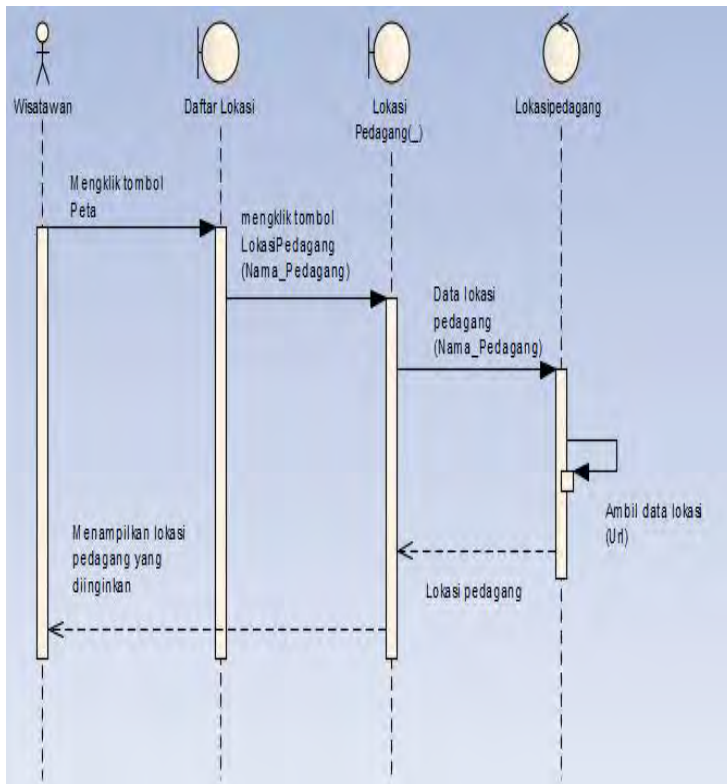


Gambar C. 2 Sequence Diagram Mencari informasi penjual.

SD02-1-01 Melihat informasi penting terkait kriminalitas

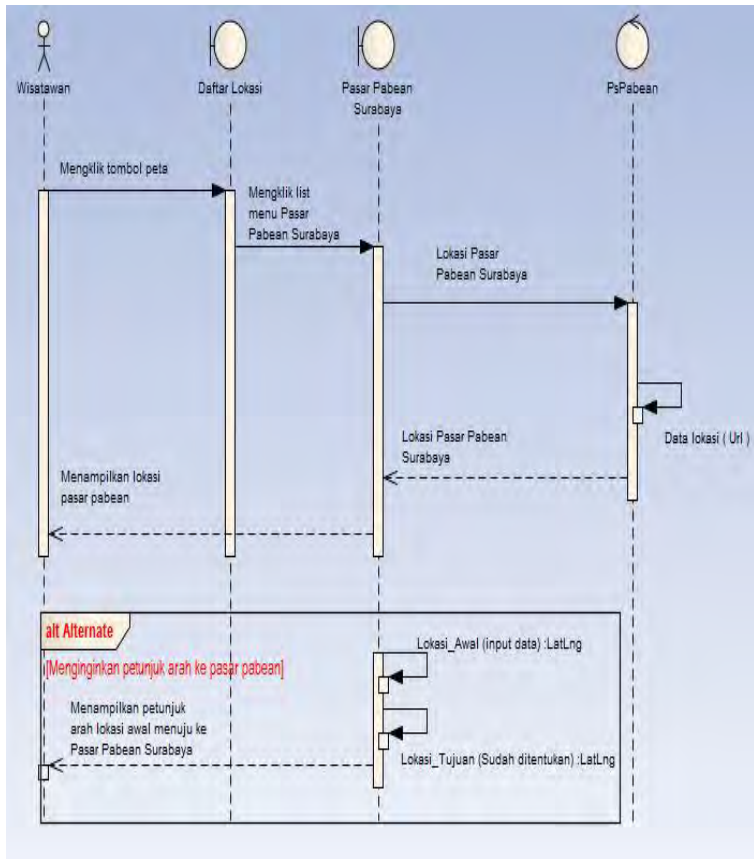


Gambar C. 3 Sequence Diagram melihat informasi penting terkait kriminalitas

SD03-1-01 Melihat titik lokasi pedagang di denah

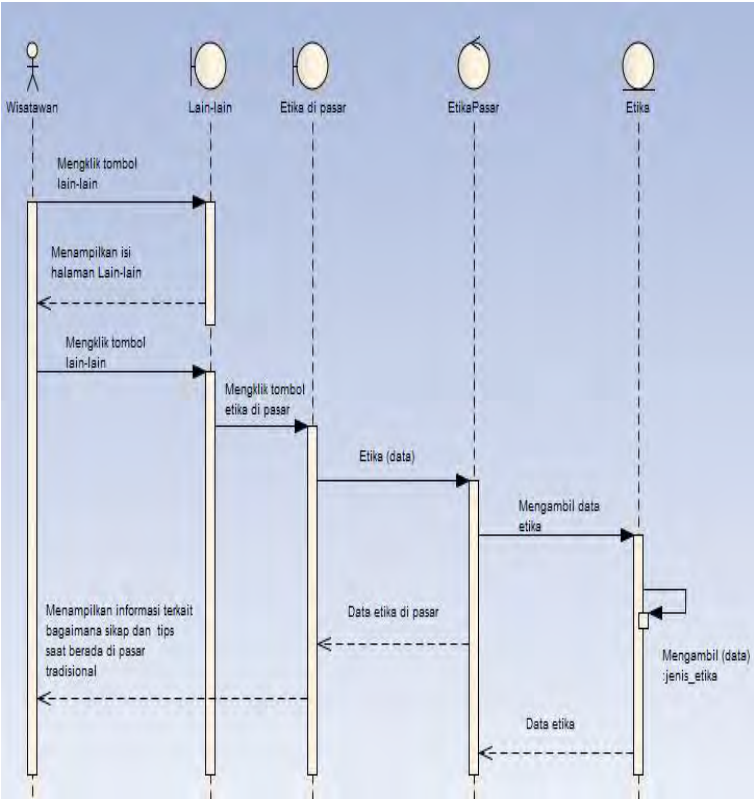
Gambar C. 4 Sequence Diagram melihat lokasi pedagang di denah

SD03-1-02 Melihat dan menentukan informasi petunjuk arah ke Pasar Pabean



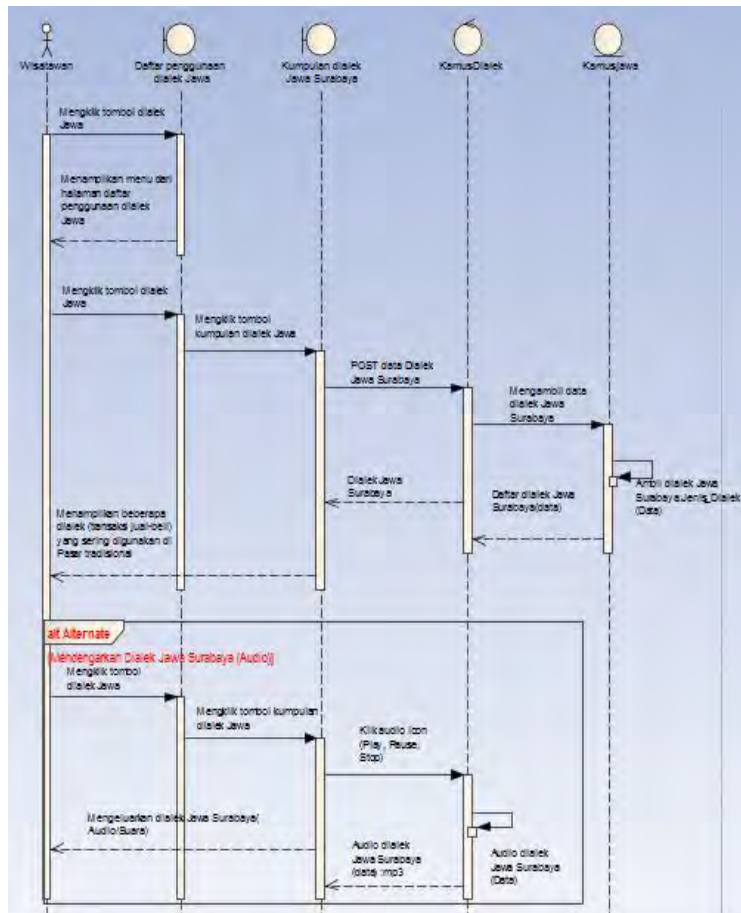
Gambar C. 5 Sequence Diagram melihat dan menentukan informasi petunjuk arah ke Pasar Pabean

SD04-1-01 Memberikan informasi etika saat di pasar



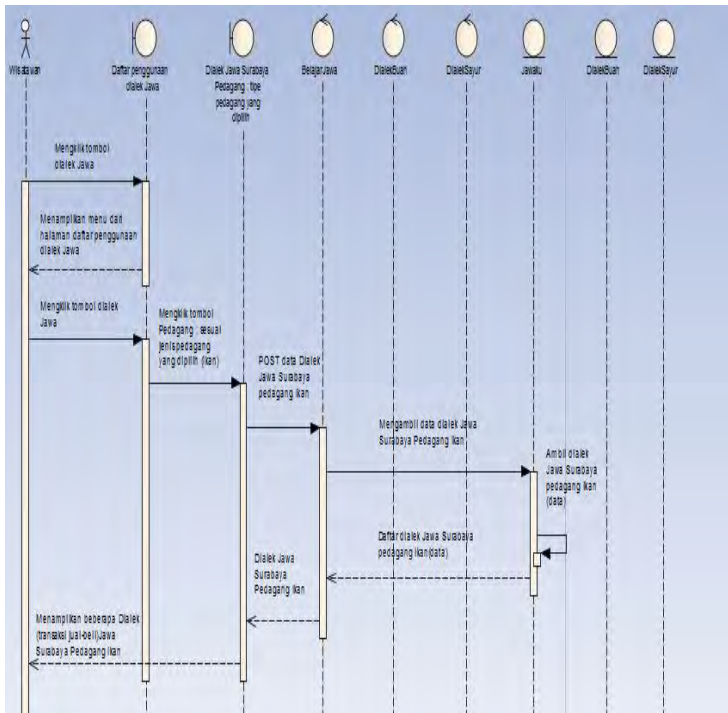
Gambar C. 6 Sequence Diagram memberikan informasi etika saat di pasar

SD05-1-01 Memberikan informasi Dialek Jawa di Pasar



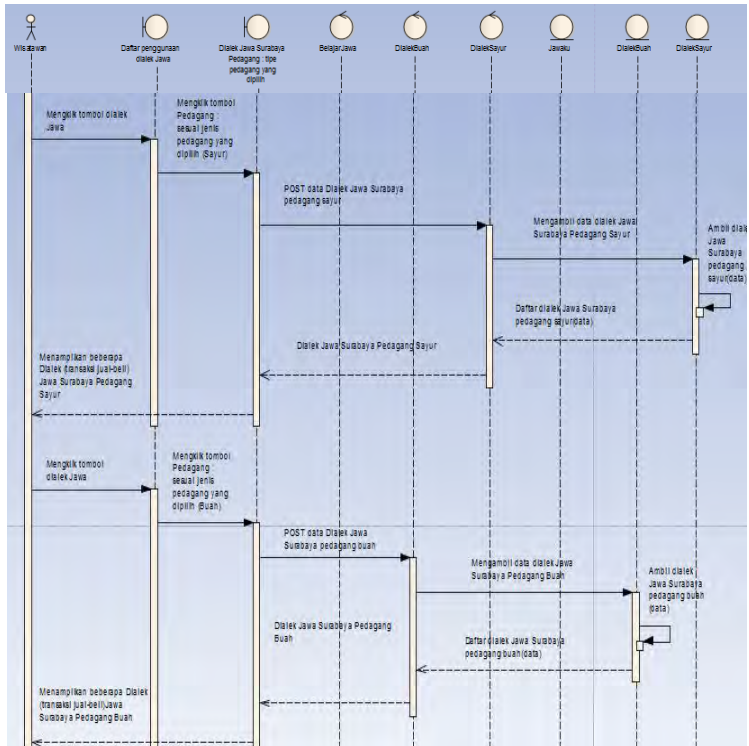
Gambar C. 7 Sequence Diagram memberikan informasi dialek Jawa di Pasar

SD05-1-02 Memberikan informasi Dialek Pasar yang dilakukan pedagang dan pembeli



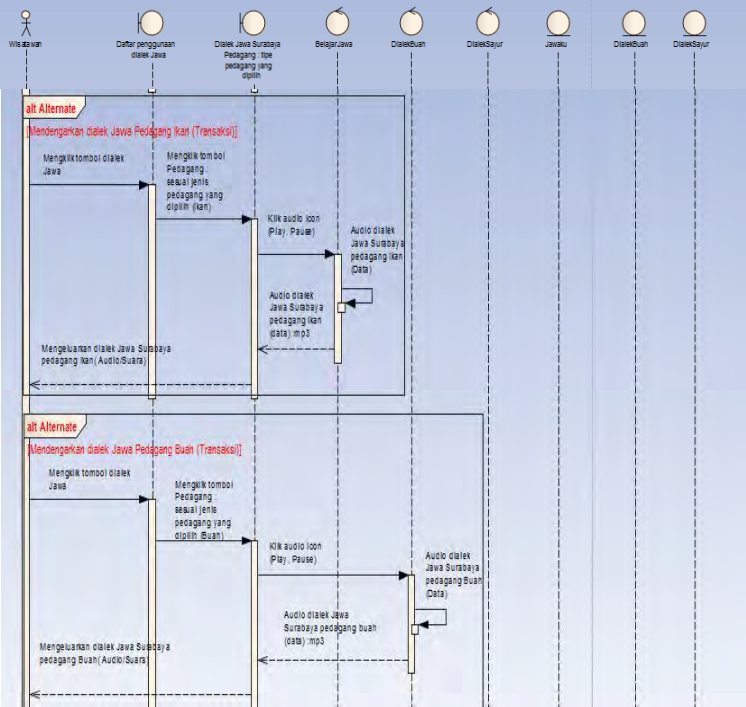
Gambar C.7.1. 1 Sequence Diagram memberikan informasi dialek pasar yang dilakukan pedagang dan pembeli

SD05-1-02 Memberikan informasi Dialek Pasar yang dilakukan pedagang dan pembeli (lanjutan)



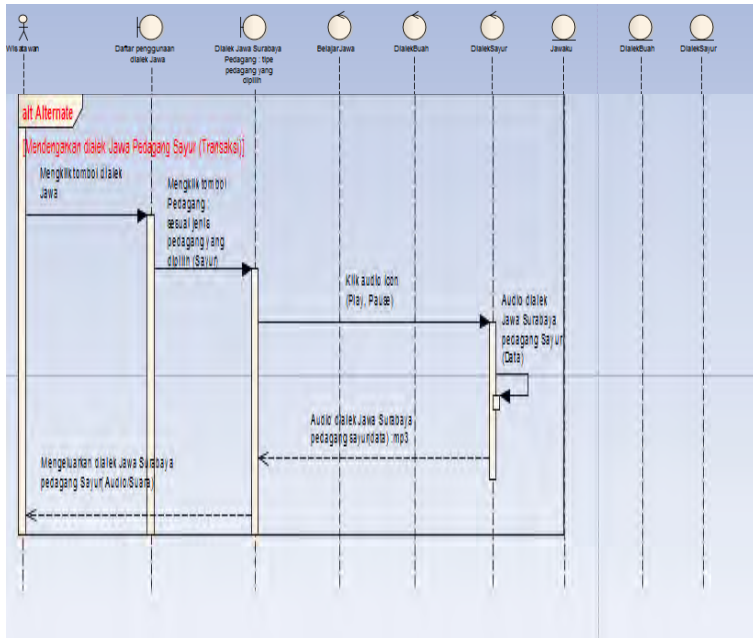
Gambar C.7.1. 2 Sequence Diagram memberikan informasi dialek pasar yang dilakukan pedagang dan pembeli

SD05-1-02 Memberikan informasi Dialek Pasar yang dilakukan pedagang dan pembeli (lanjutan)



Gambar C.7.1. 3 Sequence Diagram memberikan informasi dialek pasar yang dilakukan pedagang dan pembeli

SD05-1-02 Memberikan informasi Dialek Pasar yang dilakukan pedagang dan pembeli (lanjutan)

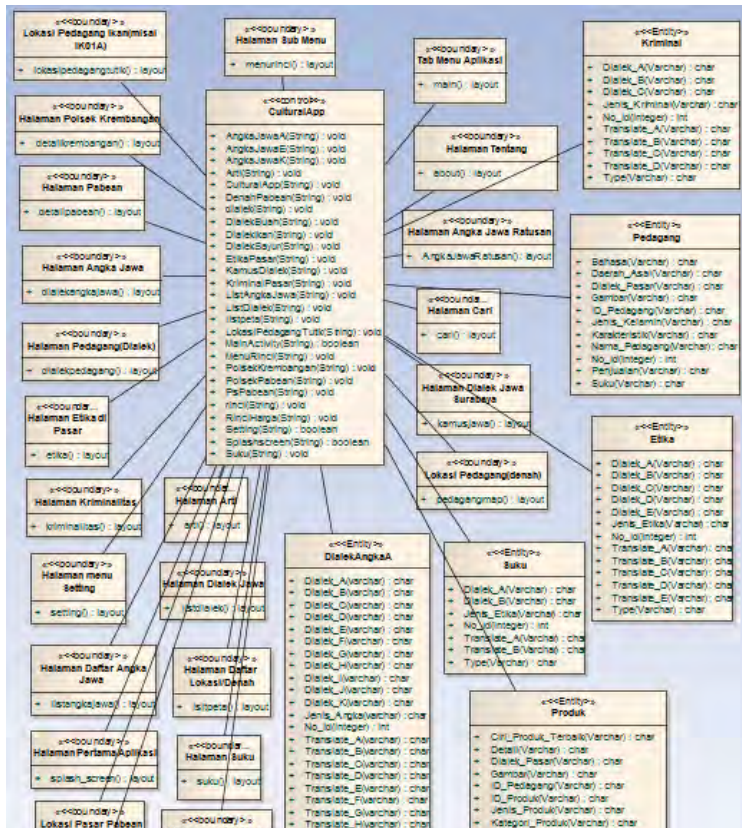


Gambar C.7.1. 4 Sequence Diagram memberikan informasi dialek pasar yang dilakukan pedagang dan pembeli

C12

(halaman ini sengaja dikosongkan)

Class Diagram Aplikasi *Cultural Experience Guide*



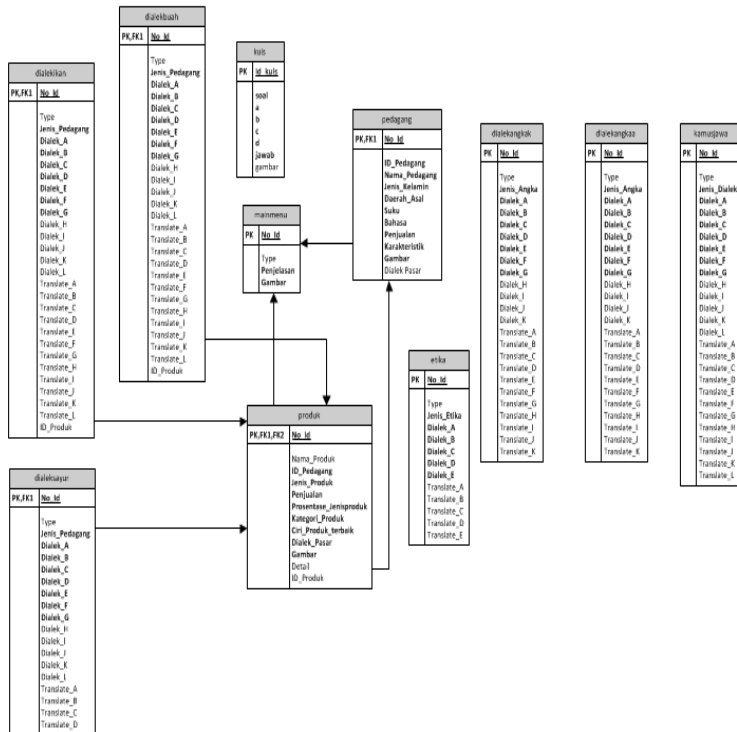
Gambar D. 1 Class Diagram Aplikasi

D2

(halaman ini sengaja dikosongkan)

LAMPIRAN E

Entity Relationship Diagram Aplikasi *Cultural Experience Guide*.



Gambar E. 1 ERD aplikasi *Cultural Experience Guide*.

(halaman ini sengaja dikosongkan)

LAMPIRAN F

User Acceptance Test (UAT)

UAT Cultural Experience Guide			
No	Task Item	Satisfy?	
		Yes	No
	Halaman Utama		
1	Splashscreen menu		
	Comment :		
2	Menampilkan menu utama		
	Comment :		
	Menu Utama		
3	Menampilkan daftar produk		
	Comment :		
4	Menampilkan daftar pedagang		
	Comment :		
5	Menampilkan dan mendengarkan dialek Jawa Surabaya		
	Comment :		
6	Menampilkan denah Pasar Pabean Surabaya		
	Comment :		
7	Menampilkan rute perjalanan menuju Pasar Pabean Surabaya.		
	Comment :		
8	Mencari informasi tiap pedagang secara rinci		
	Comment :		
9	Mencari informasi tiap produk secara rinci		
	Comment :		
10	Melihat dan mencari informasi harga komoditi terkini		

UAT <i>Cultural Experience Guide</i>			
No	Task Item	Satisfy?	
		Yes	No
	Comment :		
11	Menampilkan denah pedagang		
	Comment :		
	Sub Menu atau Lain-Lain		
12	Melihat etika di pasar		
	Comment :		
13	Melihat suku dan kebiasaan pedagang (secara garis besar)		
	Comment :		
14	Melihat informasi kriminalitas		
	Comment :		
UAT completed by:		Date completed:	
Hasil:			

LAMPIRAN G

Test Case

Tabel G. 1 Test Case Load Halaman Utama

Test Case Code		TC-01			
Test Case Name		Load halaman utama aplikasi			
No .	Aksi Test	Data Test	Hasil yang Diharapkan	Hasil Aktual	Sukses/ Gagal
1	Membuka aplikasi pertama kali.	Database sudah di <i>include</i> di Aplikasi	Database terinstall di device android	Database berhasil di Install dan halaman utama tampil	Sukses

Tabel G. 2 Test Case Melihat Daftar Produk

Test Case Code		TC-02			
Test Case Name		Melihat daftar produk			
No .	Aksi Test	Data Test	Hasil yang Diharapkan	Hasil Aktual	Sukses/ Gagal
1	Melihat daftar produk	Daftar produk yang dilihat	Menampilkan daftar produk	Aplikasi menampilkan daftar produk yang ada di Pasar Pabean Surabaya	Sukses

Tabel G. 3 Test Case Melihat Daftar Pedagang

Test Case Code		TC-03			
Test Case Name		Melihat daftar pedagang			
No .	Aksi Test	Data Test	Hasil yang Diharapkan	Hasil Aktual	Sukses /Gagal
1	Melihat daftar pedagang	Daftar pedagang yang dilihat	Menampilkan daftar pedagang	Aplikasi menampilkan daftar pedagang yang ada di Pasar Pabean Surabaya	Sukses

Tabel G. 4 Test Case Mencari Nama Produk

Test Case Code		TC-04			
Test Case Name		Mencari nama produk			
No.	Aksi Test	Data Test	Hasil yang Diharapkan	Hasil Aktual	Sukses /Gagal
1	Mengklik tab menu produk	Salah satu nama produk yang ada di daftar produk	Menampilkan hasil pencarian produk.	Menampilkan hasil pencarian nama produk	Sukses

Tabel G. 5 Test Case Mencari Nama Pedagang

Test Case Code		TC-05			
Test Case Name		Mencari nama pedagang			
No .	Aksi Test	Data Test	Hasil yang Diharapkan	Hasil Aktual	Sukses /Gagal
1	Mengklik tab menu pedagang.	Salah satu nama produk yang ada di daftar pedagang.	Menampilkan hasil pencarian pedagang.	Menampilkan hasil pencarian nama pedagang.	Sukses

Tabel G. 6 Test Case Melihat dan Mendengarkan Kumpulan Dialek Jawa

Test Case Code		TC-06			
Test Case Name		Melihat dan mendengarkan kumpulan Dialek Jawa			
No .	Aksi Test	Data Test	Hasil yang Diharapkan	Hasil Aktual	Sukses/ Gagal
1	Mengklik menu Dialek Jawa Surabaya.	List menu Kumpulan Dialek Jawa	Menampilkan Dialek Jawa Surabaya	Menampilkan Dialek Jawa Surabaya	Sukses

Test Case Code		TC-06			
Test Case Name		Melihat dan mendengarkan kumpulan Dialek Jawa			
2	Mengklik tombol play di halaman Kumpulan Dialek Jawa	Dialek Jawa Surabaya	Menghasilkan audio Dialek Jawa Surabaya	Menghasilkan suara/audio Dialek Jawa Surabaya	Sukses

Tabel G. 7 Test Case Melihat dan Mendengarkan Dialek Jawa Pedagang Ikan.

Test Case Code		TC-07			
Test Case Name		Melihat dan mendengarkan dialek Jawa Pedagang Ikan			
No .	Aksi Test	Data Test	Hasil yang Diharapkan	Hasil Aktual	Sukses/ Gagal
1	Mengklik menu Dialek Jawa Surabaya.	List menu Pedagang Ikan	Menampilkan dialek Jawa Pedagang Ikan	Menampilkan dialek Jawa Pedagang Ikan	Sukses
2	Mengklik tombol play di halaman Pedagang Ikan	Dialek Jawa Pedagang Ikan	Menghasilkan audio Dialek Jawa Pedagang Ikan	Menghasilkan suara/audio Dialek Jawa Pedagang Ikan	Sukses

Tabel G. 8 Test Case Melihat dan Mendengarkan Dialek Jawa Pedagang Buah.

Test Case Code		TC-08			
Test Case Name		Melihat dan mendengarkan dialek Jawa Pedagang Buah			
No .	Aksi Test	Data Test	Hasil yang Diharapkan	Hasil Aktual	Sukses/ Gagal
1	Mengklik menu Dialek Jawa Surabaya.	List menu Pedagang Buah	Menampilkan dialek Jawa Pedagang Buah	Menampilkan dialek Jawa Pedagang Buah	Sukses
2	Mengklik tombol play di halaman Pedagang Buah.	Dialek Jawa Pedagang Buah	Menghasilkan audio Dialek Jawa Pedagang Buah.	Menghasilkan suara/audio Dialek Jawa Pedagang Buah	Sukses

Tabel G. 9 Test Case Melihat dan Mendengarkan Dialek Jawa Pedagang Sayur.

Test Case Code		TC-09			
Test Case Name		Melihat dan mendengarkan dialek Jawa Pedagang Sayur			
No .	Aksi Test	Data Test	Hasil yang Diharapkan	Hasil Aktual	Sukses/ Gagal
1	Mengklik menu Dialek Jawa Surabaya.	List menu Pedagang Sayur.	Menampilkan dialek Jawa Pedagang Sayur.	Menampilkan dialek Jawa Pedagang Sayur.	Sukses
2	Mengklik tombol play di halaman Pedagang Sayur.	Dialek Jawa Pedagang Sayur.	Menghasilkan audio Dialek Jawa Pedagang Sayur.	Menghasilkan suara/audio Dialek Jawa Pedagang Sayur.	Sukses

Tabel G. 10 Test Case Melihat Peta Lokasi dan Petunjuk Arah ke Pasar Pabean

Test Case Code		TC-10			
Test Case Name		Melihat peta lokasi Pasar Pabean Surabaya dan mencari rute perjalanan ke Pasar Pabean Surabaya.			
No .	Aksi Test	Data Test	Hasil yang Diharapkan	Hasil Aktual	Sukses/Gagal
1	Mengklik menu peta	List menu Pasar Pabean Surabaya	Menampilkan lokasi Pasar Pabean Surabaya	Menampilkan lokasi Pasar Pabean Surabaya	Sukses
2	Menginput lokasi awal	Lokasi sesuai yang diinginkan user	Menampilkan petunjuk arah lokasi awal ke Pasar Pabean Surabaya.	Menampilkan petunjuk arah lokasi awal ke Pasar Pabean Surabaya .	Sukses

Tabel G. 11 Test Case Melihat Denah Pasar Pabean Surabaya.

Test Case Code		TC-11			
Test Case Name		Melihat denah Pasar Pabean Surabaya			
No .	Aksi Test	Data Test	Hasil yang Diharapkan	Hasil Aktual	Sukses/ Gagal
1	Mengklik menu peta	List menu denah Pasar Pabean	Menampilkan denah Pasar Pabean Surabaya.	Menampilkan denah Pasar Pabean Surabaya .	Sukses

Tabel G. 12 Melihat lokasi pedagang

Test Case Code		TC-12			
Test Case Name		Melihat lokasi pedagang			
No .	Aksi Test	Data Test	Hasil yang Diharapkan	Hasil Aktual	Sukses/ Gagal
1	Mengklik menu peta	List menu misal: Pedagang Ikan Tutik(IK01A)	Menampilkan lokasi pedagang .	Menampilkan lokasi pedagang yang diinginkan.	Sukses

Tabel G. 13 Melihat Informasi Etika di Pasar.

Test Case Code		TC-13			
Test Case Name		Melihat informasi etika di Pasar.			
No .	Aksi Test	Data Test	Hasil yang Diharapkan	Hasil Aktual	Sukses/Gagal
1	Mengklik sub menu Etika di Pasar di menu Lain-Lain.	List menu Etika di Pasar	Menampilkan tips dan informasi terkait transaksi jual-beli.	Menampilkan tips dan informasi terkait transaksi jual-beli.	Sukses

Tabel G. 14 Test Case Melihat Informasi Terkait Kriminalitas di Pasar Pabean Surabaya.

Test Case Code		TC-14			
Test Case Name		Melihat informasi terkait kriminalitas di Pasar Pabean Surabaya.			
No .	Aksi Test	Data Test	Hasil yang Diharapkan	Hasil Aktual	Sukses/Gagal
1	Mengklik sub menu Informasi Kriminalitas di menu Lain-Lain.	List menu Informasi Kriminalitas	Menampilkan tips dan informasi terkait kriminalitas.	Menampilkan tips dan informasi terkait kriminalitas.	Sukses

Tabel G. 15 Melihat Informasi Suku dan Kebiasaan di Pasar Pabean Surabaya.

Test Case Code		TC-15			
Test Case Name		Melihat informasi Suku dan Kebiasaan di Pasar Pabean Surabaya.			
No .	Aksi Test	Data Test	Hasil yang Diharapkan	Hasil Aktual	Sukses /Gagal
1	Mengklik sub menu Suku dan Kebiasaan di menu Lain-Lain.	List menu Suku dan Kebiasaan	Menampilkan tips dan informasi terkait Suku dan Kebiasaan.	Menampilkan tips dan informasi terkait Suku dan Kebiasaan	Sukses

Tabel G. 16 Test Case Melihat dan Mencari Harga Komoditi Produk di Pasar Pabean Surabaya.

Test Case Code		TC-16			
Test Case Name		Melihat dan mencari harga komoditi produk yang ada di Pasar Pabean Surabaya			
No .	Aksi Test	Data Test	Hasil yang Diharapkan	Hasil Aktual	Sukses/Gagal
1	Mengklik tombol cek harga di halaman hasil pencarian tiap pedagang.	Tidak ada karena hanya menampilkan halaman cek harga	Menampilkan halaman sistem informasi harga komoditi Pasar Pabean Surabaya.	Menampilkan halaman sistem informasi harga komoditi Pasar Pabean Surabaya .	Sukses
2	Menginput nama produk yang ingin di cek harganya.	Nama produk yang diinginkan <i>user</i> .	Menampilkan harga komoditi terkini dari produk yang diinginkan .	Menampilkan harga komoditi terkini dari produk yang diinginkan .	Sukses

Tabel G. 17 Test Case Melihat dan Mendengarkan Angka Jawa Surabaya.

Test Case Code		TC-17			
Test Case Name		Melihat dan mendengarkan angka Jawa Surabaya			
No .	Aksi Test	Data Test	Hasil yang Diharapkan	Hasil Aktual	Sukses /Gagal
1	Mengklik menu Dialek Jawa Surabaya.	List menu Angka Jawa Surabaya. Misal angka jawa 1-10	Menampilk an Angka Jawa 1-10.	Menampi lkan Angka Jawa 1-10.	Sukses
2	Mengklik tombol play di halaman Angka Jawa 1-10.	Angka Jawa 1-10.	Menghasil kan audio Angka Jawa 1-10.	Menghas ilkan suara/audio angka jawa 1-10.	Sukses

LAMPIRAN H *SOURCE CODE*

1.) Layout splashscreen.xml

```
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout
xmlns:android="http://schemas.android.com/apk/
res/android"
    android:layout_width="fill_parent"
    android:layout_height="fill_parent"
    android:background="@drawable/backgr"
    android:gravity="center_horizontal"
    android:orientation="vertical" >

    <LinearLayout

        android:id="@+id/aa"
        android:layout_width="fill_parent"
        android:layout_height="250dp"

        android:layout_alignParentBottom="true"
        android:layout_alignParentLeft="true"
        android:orientation="vertical" >

        <ProgressBar
            android:id="@+id/progg"

            style="?android:attr/progressBarStyleLarge"
            android:layout_width="40dp"
            android:layout_height="40dp"

            android:layout_gravity="center"

            android:layout_marginTop="25dp"

            android:layout_marginBottom="15dp"/>

        <TextView
```

H2

```
        android:id="@+id/state"

        android:layout_width="wrap_content"

        android:layout_height="wrap_content"

        android:layout_gravity="center"

        android:layout_marginBottom="15dp"
            android:text="Loading ..."
            android:textColor="#333"
            android:textSize="14sp" />
    </LinearLayout>

</RelativeLayout>
```

2.) Potongan kode *class* rinci

```
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
        android:layout_width="fill_parent"
        android:layout_height="fill_parent"
        android:background="@drawable/backgr"
        android:gravity="center_horizontal"
        android:orientation="vertical" >

    <LinearLayout
        android:id="@+id/linearLayout2"
        android:layout_width="fill_parent"
        android:layout_height="110dp"
        android:layout_alignParentLeft="true"
        android:layout_marginTop="60dp"

        android:background="@drawable/splash_backgroun
        d"
            android:orientation="vertical"
            android:paddingBottom="5dp"
            android:paddingTop="10dp" >
```

```

        <ImageView
            android:layout_width="wrap_content"

            android:layout_height="wrap_content"
                android:layout_gravity="center"
                android:layout_marginBottom="5dp"

            android:src="@drawable/ic_launcher" />

        <TextView
            android:id="@+id/textView1"

            android:layout_width="match_parent"

            android:layout_height="wrap_content"

            android:layout_below="@+id/linearLayout2"

            android:layout_centerHorizontal="true"
                android:layout_marginTop="4dp"
                android:gravity="center"
                android:text="Panduan Pengalaman
Kultural"

                android:textColor="#555"
                android:textSize="13sp" />

        <TextView
            android:id="@+id/textView1"

            android:layout_width="match_parent"

            android:layout_height="wrap_content"

            android:layout_below="@+id/linearLayout2"

            android:layout_centerHorizontal="true"
                android:layout_marginTop="8dp"
                android:gravity="center"
                android:text="Dalam Konteks Budaya Jawa
di Pasar Pabean"

```

```

        android:textColor="#555"
        android:textSize="13sp" />

</LinearLayout>

<LinearLayout
    android:id="@+id/aa"
    android:layout_width="fill_parent"
    android:layout_height="220dp"
    android:layout_alignParentLeft="true"

    android:layout_below="@+id/linearLayout2"
    android:orientation="vertical" >

    <Button
        android:id="@+id/dialek"
        android:layout_width="200dp"
        android:layout_height="40dp"
        android:layout_gravity="center"
        android:layout_marginBottom="0dp"
        android:layout_marginTop="10dp"
        android:text="Dialek Jawa
Surabaya" />

    <LinearLayout
        android:id="@+id/linearLayout2"

        android:layout_width="wrap_content"

        android:layout_height="wrap_content"

        android:layout_alignParentLeft="true"
        android:layout_gravity="center"
        android:layout_marginTop="0dp"
        android:orientation="horizontal"
        android:paddingBottom="0dp"
        android:paddingTop="0dp" >

        <Button
            android:id="@+id/cproduk"
            android:layout_width="100dp"

```



```

        android:layout_height="40dp"

android:layout_gravity="center"

android:layout_marginBottom="0dp"
        android:layout_marginTop="0dp"
        android:text="Produk" />

        <Button
            android:id="@+id/cpedagang"
            android:layout_width="100dp"
            android:layout_height="40dp"

android:layout_gravity="center"

android:layout_marginBottom="0dp"
        android:layout_marginTop="0dp"
        android:text="Pedagang" />
    </LinearLayout>

    <Button
        android:id="@+id/maps"
        android:layout_width="200dp"
        android:layout_height="40dp"
        android:layout_gravity="center"
        android:layout_marginBottom="0dp"
        android:layout_marginTop="0dp"
        android:text="Peta" />

    <Button
        android:id="@+id/bantuan"
        android:layout_width="200dp"
        android:layout_height="40dp"
        android:layout_gravity="center"
        android:layout_marginBottom="0dp"
        android:layout_marginTop="0dp"
        android:text="Bantuan" />

    <Button
        android:id="@+id/more"
        android:layout_width="200dp"

```

```

        android:layout_height="40dp"
        android:layout_gravity="center"
        android:layout_marginBottom="0dp"
        android:layout_marginTop="0dp"
        android:text="Lain-Lain" />
    </LinearLayout>

</RelativeLayout>

```

3.) Potongan kode *layout* main

```

<?xml version="1.0" encoding="utf-8"?>
<TabHost
xmlns:android="http://schemas.android.com/apk/res/android"
    android:id="@android:id/tabhost"
    android:layout_width="fill_parent"
    android:layout_height="fill_parent">

    <LinearLayout
        android:orientation="vertical"
        android:layout_width="fill_parent"
        android:layout_height="fill_parent"
        android:padding="5dp">

        <TabWidget
            android:id="@android:id/tabs"
            android:layout_width="fill_parent"
            android:layout_height="wrap_content" />

        <FrameLayout
            android:id="@android:id/tabcontent"
            android:layout_width="fill_parent"
            android:layout_height="fill_parent"
            android:paddingTop="5dp" />

    </LinearLayout>
</TabHost>

```

4.) Potongan kode *class* Main Activity

```
public class MainActivity extends TabActivity
{
    /** Called when the activity is first
    created. */
    private AlertDialog.Builder builder;
    private AlertDialog aboutDialog;
    final String[] tab_menu = {
        "Main Menu",
        "Produk",
        "Pedagang"};

    @Override
    public void onCreate(Bundle
savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.main);

        Resources res = getResources(); //
Resource object to get Drawables
        TabHost tabHost = getTabHost(); //
The activity TabHost
        TabHost.TabSpec spec; // Reusable
TabSpec for each tab
        Intent intent; // Reusable Intent for
each tab

        // All
        intent = new Intent().setClass(this,
rinci.class);
        intent.putExtra("DICT", 1);
        spec =
tabHost.newTabSpec("all").setIndicator(tab_men
u[0],
res.getDrawable(R.drawable.tab_style)).setCont
ent(intent);
        tabHost.addTab(spec);
```

5.) Potongan kode *class* rinci

```
public class rinci extends Activity {
    Button p ,h,t, m,d,r;
    Intent translate_dialog;
    private int DICT = 1;

    @Override
    protected void onCreate(Bundle
savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.rinci);

        /* UI Initialization */
        p = (Button)
findViewById(R.id.dialek);
        h = (Button)
findViewById(R.id.cproduk);
        t = (Button)
findViewById(R.id.cpedagang);
        m = (Button)
findViewById(R.id.maps);
        d = (Button)
findViewById(R.id.bantuan);
        r = (Button)
findViewById(R.id.more);

        d.setOnClickListener(new
OnClickListener() {

            public void onClick(View v) {
                translate("Cultural
Experience Guide");
            }
        });

        h.setOnClickListener(new
OnClickListener() {

            @Override
            public void onClick(View v) {
```

```

        translate("Daftar
Produk");

    }

});

t.setOnClickListener(new
OnClickListener() {

    @Override
    public void onClick(View v) {
        translate("Pedagang");
    }

});

m.setOnClickListener(new
OnClickListener() {

    @Override
    public void onClick(View v) {
        translates("Peta");
    }

});

p.setOnClickListener(new
OnClickListener() {

    public void onClick(View v) {
        translateku("Dialek
Jawa");
    }

});

r.setOnClickListener(new
OnClickListener() {

    public void onClick(View v) {
        translatemore("More");
    }

});

```

```

    }

    protected void translates(String text) {
        translate_dialog = new
Intent().setClass(this, listpeta.class);
        startActivity(translate_dialog);
    }

    protected void translateku(String text) {
        translate_dialog = new
Intent().setClass(this, ListDialek.class);
        startActivity(translate_dialog);
    }

    protected void translatemore(String text)
{
        translate_dialog = new
Intent().setClass(this, MenuRinci.class);
        startActivity(translate_dialog);
    }

    protected void translate(String text) {
        translate_dialog = new
Intent().setClass(this, Arti.class);
        translate_dialog.putExtra("DICT",
DICT);
        translate_dialog.putExtra("TEXT",
text);
        startActivity(translate_dialog);
    }
}

```

6.) Potongan kode *class* SplashScreen.

```

public class SplashScreen extends Activity {

    ProgressBar progressBar;
    TextView state;
    Handler handler = new Handler();
    DBKamus db_kamus_helper;

```

```

@Override
    public void onCreate(Bundle
savedInstanceState) {
        super.onCreate(savedInstanceState);

        setContentView(R.layout.splash_screen);
        db_kamus_helper = new
DBKamus(SplashScreen.this);
        state = (TextView)
findViewById(R.id.state);
        InitDatabase();
    }

    @Override
    public void
onConfigurationChanged(Configuration
newConfig) {

        super.onConfigurationChanged(newConfig);
    }

    @Override
    public boolean onKeyDown(int keyCode,
KeyEvent event) {
        if (keyCode == KeyEvent.KEYCODE_BACK
&& event.getRepeatCount() == 0) {
            return true;
        }
        return super.onKeyDown(keyCode,
event);
    }

    public void InitDatabase() {
        new Thread(new Runnable() {
            @Override
            public void run() {
                delay(1200);
                if
(!db_kamus_helper.isDatabaseExist()) {
                    handler.post(new Runnable() {

```

H12

```
        @Override
        public void run() {
state.setText("Menginstal Database ...");
        }
    });
    try
{db_kamus_helper.createDataBase();} catch
(IOException e) {e.printStackTrace();};
    }
    handler.post(new Runnable() {
        @Override
        public void run() {
            state.setText("Memulai
...");
        }
    });
    delay(800);
    handler.post(new Runnable() {
        @Override
        public void run() {
// TODO Auto-generated
method stub
            Intent main_intent = new
Intent(SplashScreen.this, MainActivity.class);

SplashScreen.this.startActivity(main_intent);
            SplashScreen.this.finish();
        }
    });
}

    public void delay(int millis) {
        try{
            Thread.sleep(millis);
        }catch(InterruptedException ie){
            ie.printStackTrace();
        }
    }
    }).start();
}
}
```


7.) Potongan kode *class* rinci menampilkan informasi produk.

```
public class rinci extends Activity {
    Button p ,h,t, m,d,r;
    Intent translate_dialog;
    private int DICT = 1;

    @Override
    protected void onCreate(Bundle
savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.rinci);

        /* UI Initialization */
        p = (Button)
findViewById(R.id.dialek);
        h = (Button)
findViewById(R.id.cproduk);
        t = (Button)
findViewById(R.id.cpedagang);
        m = (Button)
findViewById(R.id.maps);
        d = (Button)
findViewById(R.id.bantuan);
        r = (Button)
findViewById(R.id.more);

        d.setOnClickListener(new
OnClickListener() {

            public void onClick(View v) {
                translate("Cultural
Experience Guide");
            }
        });

        h.setOnClickListener(new
OnClickListener() {

            @Override
```

```

        public void onClick(View v) {
            translate("Daftar
Produk");
        }
    });

    t.setOnClickListener(new
OnClickListener() {

        @Override
        public void onClick(View v) {
            translate("Pedagang");
        }
    });

    m.setOnClickListener(new
OnClickListener() {

        @Override
        public void onClick(View v) {
            translates("Peta");
        }
    });

    p.setOnClickListener(new
OnClickListener() {

        public void onClick(View v) {
            translateku("Dialek
Jawa");
        }
    });

    r.setOnClickListener(new
OnClickListener() {

        public void onClick(View v) {
            translatemore("More");
        }
    });

```

```

    }

    protected void translates(String text) {
        translate_dialog = new
Intent().setClass(this, listpeta.class);
        startActivity(translate_dialog);
    }

    protected void translateku(String text) {
        translate_dialog = new
Intent().setClass(this, ListDialek.class);
        startActivity(translate_dialog);
    }

    protected void translatemore(String text)
{
        translate_dialog = new
Intent().setClass(this, MenuRinci.class);
        startActivity(translate_dialog);
    }

    protected void translate(String text) {
        translate_dialog = new
Intent().setClass(this, Arti.class);
        translate_dialog.putExtra("DICT",
DICT);
        translate_dialog.putExtra("TEXT",
text);
        startActivity(translate_dialog);
    }
}

```

8.) Potongan kode *class* Arti menampilkan informasi produk.

```

public class Arti extends Activity implements
TextToSpeech.OnInitListener {

    final static int SEMUA = 1;

```

```

    final static int    HEWAN = 2;
    final static int    TUMBUHAN = 3;
    final String[] dict_name = new String[]
{"Semua", "HEWAN", "TUMBUHAN"};

    private int DICT;
    private String text;
    private SQLiteDatabase db = null;
    private Cursor kingdom = null;
    private DBKamus db_kamus_helper;
    private TextToSpeech mTts;
    private int SpeechStatus;
    private float font_size;
    Intent translate_dialog;

    TextView title, content, detail;
    Button speech, close, audio, harga;

    WebView pic;

    Handler handler = new Handler();
    Thread searching;

    @Override
    public void onCreate(Bundle
savedInstanceState) {
        super.onCreate(savedInstanceState);

        /* Get Intent Parameter */
        this.getIntentParameter();

        /* Open Database File */
        db_kamus_helper = new DBKamus(this);
        try {
            db_kamus_helper.openDataBase();
            db =
db_kamus_helper.getReadableDatabase();
        } catch (SQLException sqle) {
            throw sqle;
        }

        /* UI Initialization */

```

```

        if (checkResult()) this.initUI();
    }

    private void getIntentParameter() {
        // TODO Auto-generated method stub
        Uri data = this.getIntent().getData();
        if (data!=null) {
            DICT =
Integer.parseInt(data.getQueryParameter("dict"
));
            text =
data.getQueryParameter("text");
        } else {
            DICT =
this.getIntent().getExtras().getInt("DICT",
1);
            text =
this.getIntent().getExtras().getString("TEXT")
;
        }
    }

    private boolean checkResult() {
        // TODO Auto-generated method stub
        String kingdom1;
        switch(DICT) {
            default:
            case SEMUA:
                kingdom1 = "SELECT * FROM
MainMenu where
lower(Type)=lower('"+text.replace("'",
"''")+"' ) ";
                break;
            case HEWAN:
                kingdom1 = "SELECT * FROM
Produk where
lower>Nama_Produk)=lower('"+text.replace("'",
"''")+"' ) ";
                break;
            case TUMBUHAN:

```

```

        kingdom1 = "SELECT * FROM
Pedagang where
lower(ID_Pedagang)=lower('"+text.replace("'",
"''")+"'");
        break;
    }
    kingdom = db.rawQuery(kingdom1, null);
    if (!kingdom.moveToFirst()) {
        Toast.makeText(this, "word
`"+text+"` not found in "+dict_name[DICTIONARY-1]+"
dictionary!", Toast.LENGTH_SHORT).show();
        this.finish();
        return false;
    } else return true;
}

private void initUI() {
    // TODO Auto-generated method stub

this.requestWindowFeature(Window.FEATURE_NO_TITLE);
    this setContentView(R.layout.arti);

this.getWindow().setLayout(LayoutParams.FILL_PARENT,
LayoutParams.FILL_PARENT);
    mTts = new TextToSpeech(this, this);

    title = (TextView)
findViewById(R.id.title);
    content = (TextView)
findViewById(R.id.content);
    detail = (TextView)
findViewById(R.id.detail);
    speech = (Button)
findViewById(R.id.speech);
    close = (Button)
findViewById(R.id.close);
    harga = (Button)
findViewById(R.id.harga);
    //audio = (Button)
findViewById(R.id.audio);

```

```

        pic = (WebView)
findViewById(R.id.tampil);

        /* Define event callback function */
        speech.setOnClickListener(new
View.OnClickListener() {
            @Override
            public void onClick(View arg0) {
                // TODO Auto-generated method
stub
                if (SpeechStatus ==
TextToSpeech.LANG_MISSING_DATA || SpeechStatus
== TextToSpeech.LANG_NOT_SUPPORTED)
                    installLanguage();
                else
                    mTts.speak(text,
TextToSpeech.QUEUE_FLUSH, null);
            }
        });

        close.setOnClickListener(new
View.OnClickListener() {
            @Override
            public void onClick(View arg0) {
                // TODO Auto-generated method
stub
                Arti.this.finish();
            }
        });

        harga.setOnClickListener(new
View.OnClickListener() {

            public void onClick(View arg0) {
                translate("harga");
            }
        });
    }
}

```

```

private boolean isValidUrl(String url) {
    Pattern p = Patterns.WEB_URL;
    Matcher m = p.matcher(url);
    if(m.matches())
        return true;
    else
        return false;
}

protected void translate(String text) {
    translate_dialog = new
Intent().setClass(this, RinciHarga.class);
    startActivity(translate_dialog);

}
@Override
public void onStart() {
    super.onStart();

    /* Get Preferences */
    SharedPreferences prefs =
PreferenceManager.getDefaultSharedPreferences(
getBaseContext());
    font_size =
Float.parseFloat(prefs.getString("font_size",
"18"));

    /* Display Result */
    title.setText(text);

content.setTextSize(TypedValue.COMPLEX_UNIT_SP
, font_size);
    switch(DICT){
        case SEMUA :
            if(kingdom.getString(3).equals("-
")){
pic.loadUrl("file:///android_asset/img/Q.jpg")
;

```



```

pic.getSettings().setJavaScriptEnabled(true);

        }else{

pic.loadUrl(kingdom.getString(3));
        }

content.setText("Penjelasan\t:\t"+kingdom.getSt
tring(2));

content.setLinkTextColor(Color.rgb(0x44, 0x44,
0xBB));

        break;

        case HEWAN :
            if(kingdom.getString(9).equals("-
")){

pic.loadUrl("file:///android_asset/img/Q.jpg")
;

pic.getSettings().setJavaScriptEnabled(true);
        }else{

pic.loadUrl(kingdom.getString(9));
        }

detail.setText(kingdom.getString(10));
        content.setText("Dialek
Pasar\t:\t"+kingdom.getString(8)+
        "\nID
Pedagang\t:" +kingdom.getString(2)+
        "\nJenis
Produk\t:\t"+kingdom.getString(3)+
        "\nPenjualan Per
Hari (Kg)\t:\t"+kingdom.getString(4)+

```

```

        "\nPenjualan Jenis
Produk(%) \t:\t"+kingdom.getString(5)+
        "\nKategori
Produk\t:\t"+kingdom.getString(6)+
        "\nCiri Produk
Terbaik\t:\t"+kingdom.getString(7));

content.setLinkTextColor(Color.rgb(0x44, 0x44,
0xBB));

        break;

        case TUMBUHAN :
            if(kingdom.getString(9).equals("-
")){
pic.loadUrl("file:///android_asset/img/Q.jpg")
;

pic.getSettings().setJavaScriptEnabled(true);
            }else{
pic.loadUrl(kingdom.getString(9));
            }

            content.setText("Dialek Jual
Beli\t:\t"+kingdom.getString(10)+
            "\nNama
Pedagang\t:\t"+kingdom.getString(2)+
            "\nJenis
Kelamin\t:\t"+kingdom.getString(3)+
            "\nDaerah
Asal\t:\t"+kingdom.getString(4)+
            "\nSuku\t:\t"+kingdom.getString(5)+
            "\nBahasa yang digunakan
transaksi\t:\t"+kingdom.getString(6)+
            "\nPenjualan laku tiap
hari(Kg) \t:\t"+kingdom.getString(7)+
            "\nKarakteristik
Pedagang\t:\t"+kingdom.getString(8));

```

```

content.setLinkTextColor(Color.rgb(0x44, 0x44,
0xBB));
                break;}
}

```

9.) Potongan kode *layout* arti.

```

<?xml version="1.0" encoding="utf-8"?>
<LinearLayout
xmlns:android="http://schemas.android.com/apk/
res/android"
    android:layout_width="fill_parent"
    android:layout_height="fill_parent"
    android:background="@drawable/background"
    android:orientation="vertical"
    android:paddingBottom="15dp"
    android:paddingLeft="10dp"
    android:paddingRight="10dp"
    android:paddingTop="7dp" >

    <RelativeLayout
    android:id="@+id/titlebar"
    android:layout_width="fill_parent"
    android:layout_height="wrap_content"
    android:paddingBottom="4dp">

    <Button
    android:id="@+id/speech"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"

    android:background="@drawable/speech_button"
    android:layout_gravity="center_horizontal"
    android:layout_alignParentTop="true"
    android:layout_toLeftOf="@+id/separator">
    </Button>

    <View
    android:id="@+id/separator"

```

```

        android:layout_width="4dp"
        android:layout_height="4dp"
        android:background="#FAFAFA"
        android:layout_gravity="center_horizontal"
        android:layout_alignParentTop="true"
        android:layout_toLeftOf="@+id/close"
    />
    <Button
        android:id="@+id/close"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:background="@drawable/close_button"
        android:layout_alignParentTop="true"
        android:layout_alignParentRight="true">
    </Button>
</RelativeLayout>
<View
    android:id="@+id/horizontalLine"
    android:layout_width="fill_parent"
    android:layout_height="2dip"
    android:background="#888"
/>

<TextView
    android:id="@+id/title"
    android:layout_width="fill_parent"
    android:layout_height="wrap_content"
    android:textColor="#F8F8FF"
    android:textSize="22sp"
    android:textStyle="italic"
    android:paddingBottom="5dp">
</TextView>

<WebView
    android:id="@+id/tampil"
    android:layout_width="138dp"
    android:layout_height="133dp"
    android:layout_gravity="center"

```

```

        android:layout_marginBottom="20dp"
        android:layout_marginTop="10dp" />

<TextView
    android:id="@+id/detail"
    android:layout_width="fill_parent"
    android:layout_height="wrap_content"
    android:textColor="#FF0000"
    android:textSize="18sp"
    android:textStyle="italic"
    android:paddingBottom="5dp">
</TextView>

<ScrollView
    android:id="@+id/content_view"
    android:layout_width="fill_parent"
    android:layout_height="fill_parent">

    <LinearLayout

        android:layout_width="wrap_content"

        android:layout_height="wrap_content"
            android:orientation="vertical" >

        <TextView
            android:id="@+id/content"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:background="@drawable/bg"
            android:textColor="#FFFFFF" />

        <Button
            android:id="@+id/harga"

            android:layout_width="wrap_content"

            android:layout_height="wrap_content"

            android:background="@drawable/play"
            android:textColor="#FFFFFF"

```

```

        android:text="@string/cek" />

    </LinearLayout>
</ScrollView>
</LinearLayout>

```

10.) Potongan kode *class* arti untuk menampilkan detail produk.

```

private boolean checkResult() {
    // TODO Auto-generated method stub
    String kingdom1;
    switch(DICT) {
        default:
            case HEWAN:
                kingdom1 = "SELECT * FROM
Produk where
lower>Nama_Produk)=lower('"+text.replace("'",
"''")+"'");
                break;
    }
}

public void onStart() {
    super.onStart();

    /* Get Preferences */
    SharedPreferences prefs =
PreferenceManager.getDefaultSharedPreferences(
getBaseContext());
    font_size =
Float.parseFloat(prefs.getString("font_size",
"18"));

    /* Display Result */
    title.setText(text);

    content.setTextSize(TypedValue.COMPLEX_UNIT_SP
, font_size);
    switch(DICT){
        case HEWAN :

```

```

        if (kingdom.getString(9).equals("-")) {
pic.loadUrl("file:///android_asset/img/Q.jpg");
;

pic.getSettings().setJavaScriptEnabled(true);
        } else {

pic.loadUrl(kingdom.getString(9));
        }

detail.setText(kingdom.getString(10));
        content.setText("Dialek
Pasar\t:\t"+kingdom.getString(8)+
        "\nID
Pedagang\t:" +kingdom.getString(2)+
        "\nJenis
Produk\t:\t"+kingdom.getString(3)+
        "\nPenjualan Per
Hari (Kg)\t:\t"+kingdom.getString(4)+
        "\nPenjualan Jenis
Produk(%) \t:\t"+kingdom.getString(5)+
        "\nKategori
Produk\t:\t"+kingdom.getString(6)+
        "\nCiri Produk
Terbaik\t:\t"+kingdom.getString(7));

content.setLinkTextColor(Color.rgb(0x44, 0x44,
0xBB));

        break;
    }
}

```

11.) Potongan kode *class* MainActivity untuk menampilkan tab menu pedagang.

```

public class MainActivity extends TabActivity
{
    /** Called when the activity is first
created. */
    private AlertDialog.Builder builder;
    private AlertDialog aboutDialog;
    final String[] tab_menu = {
        "Main Menu",
        "Produk",
        "Pedagang"};

    @Override
    public void onCreate(Bundle
savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.main);

        Resources res = getResources(); //
Resource object to get Drawables
        TabHost tabHost = getTabHost(); //
The activity TabHost
        TabHost.TabSpec spec; // Resusable
TabSpec for each tab
        Intent intent; // Reusable Intent for
each tab

        // All
        intent = new Intent().setClass(this,
rinci.class);
        intent.putExtra("DICT", 1);
        spec =
tabHost.newTabSpec("all").setIndicator(tab_men
u[0],
res.getDrawable(R.drawable.tab_style)).setCont
ent(intent);
        tabHost.addTab(spec);

        // English to Indonesian
        intent = new Intent().setClass(this,
Cari.class);
        intent.putExtra("DICT", 2);

```



```

        spec =
tabHost.newTabSpec("en2id").setIndicator(tab_m
enu[1],
res.getDrawable(R.drawable.tab_style)).setCont
ent(intent);
        tabHost.addTab(spec);

        // Indonesian to English
        intent = new Intent().setClass(this,
Cari.class);
        intent.putExtra("DICT", 3);
        spec =
tabHost.newTabSpec("id2en").setIndicator(tab_m
enu[2],
res.getDrawable(R.drawable.tab_style)).setCont
ent(intent);
        tabHost.addTab(spec);

```

12.) Potongan kode *class cari.java* pencarian pedagang/penjual.

```

protected void cari(String text) {
    // TODO Auto-generated method stub
    list_result.clear();
    if (text.length() > 0) {
        String query;
        switch (DICT) {
            default:
            case SEMUA:
                query = "SELECT
Nama_Produk as word FROM Produk where word
LIKE '"+text.replace("'", "''")+"%' UNION
SELECT ID_Pedagang as word FROM Pedagang where
word LIKE '"+text.replace("'", "''")+"%' ORDER
BY word LIMIT "+search_limit;
                break;
            case HEWAN:
                query = "SELECT
Nama_Produk FROM Produk where Nama_Produk LIKE

```

```

        "+text.replace("'", "''")+%"' ORDER BY
Nama_Produk LIMIT "+search_limit;
        break;
        case TUMBUHAN:
            query = "SELECT
ID_Pedagang FROM Pedagang where ID_Pedagang
LIKE '"+text.replace("'", "''")+%"' ORDER BY
ID_Pedagang LIMIT "+search_limit;
            break;
        }
        kamusCursor = db.rawQuery(query,
null);
        if (kamusCursor.moveToFirst()) {
list_result.add(kamusCursor.getString(0));
            while(!kamusCursor.isLast()) {
                kamusCursor.moveToNext();

list_result.add(kamusCursor.getString(0));
            }
        } else {
            list_result.add(not_found);
        }
    }
}

protected void translate(String text) {
    translate_dialog = new
Intent().setClass(this, Arti.class);
    translate_dialog.putExtra("DICT",
DICT);
    translate_dialog.putExtra("TEXT",
text);
    startActivity(translate_dialog);
}
}

```

13.) Potongan kode *detail* pedagang/penjual.

```

private boolean checkResult() {
    // TODO Auto-generated method stub

```

```

        String kingdom1;
        switch(DICT) {
            default:
            case TUMBUHAN:
                kingdom1 = "SELECT * FROM
Pedagang where
lower(ID_Pedagang)=lower('"+text.replace("'",
"''")+"'");
                break;
        }

public void onStart() {
    super.onStart();

    /* Get Preferences */
    SharedPreferences prefs =
PreferenceManager.getDefaultSharedPreferences(
getBaseContext());
    font_size =
Float.parseFloat(prefs.getString("font_size",
"18"));

    /* Display Result */
    title.setText(text);

    content.setTextSize(TypedValue.COMPLEX_UNIT_SP
, font_size);
    switch(DICT){
        case TUMBUHAN :
            if(kingdom.getString(9).equals("-
")){

pic.loadUrl("file:///android asset/img/Q.jpg")
;

pic.getSettings().setJavaScriptEnabled(true);
            }else{

pic.loadUrl(kingdom.getString(9));
            }

```

```

        content.setText("Dialek Jual
Beli\t:\t"+kingdom.getString(10)+
        "\nNama
Pedagang\t:\t"+kingdom.getString(2)+
        "\nJenis
Kelamin\t:\t"+kingdom.getString(3)+
        "\nDaerah
Asal\t:\t"+kingdom.getString(4)+

        "\nSuku\t:\t"+kingdom.getString(5)+
        "\nBahasa yang digunakan
transaksi\t:\t"+kingdom.getString(6)+
        "\nPenjualan laku tiap
hari (Kg)\t:\t"+kingdom.getString(7)+
        "\nKarakteristik
Pedagang\t:\t"+kingdom.getString(8));

content.setLinkTextColor(Color.rgb(0x44, 0x44,
0xBB));

        break;
    }

}

```

14.) Potongan kode *class* listpeta.

```

public class listpeta extends ListActivity {

    @Override
    protected void onCreate(Bundle icicle) {
        super.onCreate(icicle);
        setContentView(R.layout.listpeta);
        String[] menulokasi = new
String[]{"Pasar Pabean Surabaya", "Denah Pasar
Pabean", "Pedagang Ikan Tutik(IK01A)"};

        this.setAdapter(new
ArrayAdapter<String>(this, android.R.layout.sim
ple_list_item_1, menulokasi));
    }
}

```

```

//defenisi untuk handle array di click
protected void onListItemClick(ListView
l,View v,int position,long id)
{
    super.onListItemClick(l, v, position,
id);
    // menangkap string yang click
    Object o =
this.getListAdapter().getItem(position);
    String pilihan = o.toString();
    tampilkanpilihan(pilihan);
}

private void tampilkanpilihan(String
pilihan) {
    // TODO Auto-generated method stub
    try
    {
        Intent i = null;
        if(pilihan.equals("Pasar Pabean
Surabaya"))
        {
            i = new Intent
(this,PsPabean.class);
        }
        if(pilihan.equals("Denah Pasar
Pabean"))
        {
            i = new Intent
(this,DenahPabean.class);
        }

        if(pilihan.equals("Pedagang Ikan
Tutik(IK01A)"))
        {
            i = new Intent
(this,LokasiPedagangTutik.class);
        }
    }
}

```

```

        startActivity(i);

    } catch (Exception e)
    {
        e.printStackTrace();
    }

}

}

```

15.) Potongan kode informasi pedagang (Salah satu contoh pedagang Ikan Tutik IK01A).

```

public class LokasiPedagangTutik extends
Activity {

    // These matrices will be used to move and
    zoom image
    Matrix matrix = new Matrix();
    Matrix savedMatrix = new Matrix();

    // We can be in one of these 3 states
    static final int NONE = 0;
    static final int DRAG = 1;
    static final int ZOOM = 2;
    int mode = NONE;

    // Remember some things for zooming
    PointF start = new PointF();
    PointF mid = new PointF();
    float oldDist = 1f;
    String savedItemClicked;
    ImageView imageView1;

    String TAG = "DEBUG";

    @Override
    protected void onCreate(Bundle
savedInstanceState) {
        super.onCreate(savedInstanceState);

```

```

setContentView(R.layout.lokasipedagangtutik);

        imageView1 =
(ImageView) findViewById(R.id.imageView1);

        imageView1.setOnTouchListener(new
OnTouchListener() {

            @Override
            public boolean onTouch(View v,
MotionEvent event) {
                // TODO Auto-generated method
stub
                ImageView view = (ImageView)
v;

                dumpEvent(event);

                // Handle touch events here...
                switch (event.getAction() &
MotionEvent.ACTION_MASK) {
                    case MotionEvent.ACTION_DOWN:
                        savedMatrix.set(matrix);
                        start.set(event.getX(),
event.getY());
                        Log.d(TAG, "mode=DRAG");
                        mode = DRAG;
                        break;
                    case
MotionEvent.ACTION_POINTER_DOWN:
                        oldDist = spacing(event);
                        Log.d(TAG, "oldDist=" +
oldDist);
                        if (oldDist > 10f) {
                            savedMatrix.set(matrix);
                            midPoint(mid, event);
                            mode = ZOOM;
                            Log.d(TAG,
"mode=ZOOM");
                        }
                }
            }
        }
    }

```

```

        break;
        case MotionEvent.ACTION_UP:
        case
MotionEvent.ACTION_POINTER_UP:
            if ((event.getX() ==
start.x) && (event.getY() == start.y))
            {
                //KHUSUS UNTUK KLIK
                (TOUCH SAJA) TANPA DRAG ATAU ZOOM

                //Toast.makeText(getApplicationContext(),
                "TOUCH", Toast.LENGTH_SHORT).show();
                float eventX =
event.getX();
                float eventY =
event.getY();
                float[] eventXY = new
float[] {eventX, eventY};

                Matrix invertMatrix =
new Matrix();

                ((ImageView) imageView1).getImageMatrix().invert(
invertMatrix);

                invertMatrix.mapPoints(eventXY);
                int px =
Integer.valueOf((int) eventXY[0]);
                int py =
Integer.valueOf((int) eventXY[1]);

                //touchedXY.setText(
                // "touched position:
"
                // +
String.valueOf(eventX) + " / "
                // +
String.valueOf(eventY));
                //invertedXY.setText(
                // "touched position:
"

```



```

// +
String.valueOf(x) + " / "
// +
String.valueOf(y));

Drawable imgDrawable =
((ImageView) imageView1).getDrawable();
Bitmap bitmap =
((BitmapDrawable) imgDrawable).getBitmap();

//imgSize.setText(
// "drawable size: "
// +
String.valueOf(bitmap.getWidth()) + " / "
// +
String.valueOf(bitmap.getHeight()));

//Limit x, y range
within bitmap
if(px < 0){
    px = 0;
}else if(px >
bitmap.getWidth()-1){
    px =
bitmap.getWidth()-1;
}

if(py < 0){
    py = 0;
}else if(py >
bitmap.getHeight()-1){
    py =
bitmap.getHeight()-1;
}

//double pxx = (px /
bitmap.getWidth()) * 100;
//double pyy = (py /
bitmap.getHeight()) * 100;
double pxx = (px *
100);

```

```

        double pww =
bitmap.getWidth();
        pxx = pxx / pww;
        double pyy = (py *
100);
        double phh =
bitmap.getWidth();
        pyy = pyy / phh;

Toast.makeText(getApplicationContext(),
"TOUCH- " + pxx + " : " + pyy,
Toast.LENGTH_SHORT).show();

    }
    mode = NONE;
    Log.d(TAG, "mode=NONE");
    break;
    case MotionEvent.ACTION_MOVE:
        if (mode == DRAG) {
            // ...

matrix.set(savedMatrix);

matrix.postTranslate(event.getX() - start.x,
event.getY()
                    - start.y);
        } else if (mode == ZOOM) {
            float newDist =
spacing(event);
            Log.d(TAG, "newDist="
+ newDist);
            if (newDist > 10f) {
matrix.set(savedMatrix);
                float scale =
newDist / oldDist;
matrix.postScale(scale, scale, mid.x, mid.y);
            }
        }
        break;

```

```

    }

    view.setImageMatrix(matrix);

    return true;
}

});
}

private void dumpEvent(MotionEvent event)
{
    String names[] = { "DOWN", "UP",
"MOVE", "CANCEL", "OUTSIDE",
"POINTER_DOWN", "POINTER_UP",
"7?", "8?", "9?" };
    StringBuilder sb = new
StringBuilder();
    int action = event.getAction();
    int actionCode = action &
MotionEvent.ACTION_MASK;
    sb.append("event
ACTION_").append(names[actionCode]);
    if (actionCode ==
MotionEvent.ACTION_POINTER_DOWN
        || actionCode ==
MotionEvent.ACTION_POINTER_UP) {
        sb.append("(pid ").append(
            action >>
MotionEvent.ACTION_POINTER_ID_SHIFT);
        sb.append(")");
    }
    sb.append("[");
    for (int i = 0; i <
event.getPointerCount(); i++) {
        sb.append("#").append(i);
        sb.append("(pid
").append(event.getPointerId(i));
        sb.append(")=").append((int)
event.getX(i));
    }
}

```

```

        sb.append(", ").append((int)
event.getY(i));
        if (i + 1 <
event.getPointerCount())
            sb.append(";");
    }
    sb.append("]");
    Log.d(TAG, sb.toString());
}

/** Determine the space between the first
two fingers */
private float spacing(MotionEvent event) {
    float x = event.getX(0) -
event.getX(1);
    float y = event.getY(0) -
event.getY(1);
    return FloatMath.sqrt(x * x + y * y);
}

/** Calculate the mid point of the first
two fingers */
private void midPoint(PointF point,
MotionEvent event) {
    float x = event.getX(0) +
event.getX(1);
    float y = event.getY(0) +
event.getY(1);
    point.set(x / 2, y / 2);
}
}

```

16.) Potongan kode *class* EtikaPasar.

```

public class EtikaPasar extends Activity {

    private SQLiteDatabase db = null;
    private TextView judul;
    private TextView jawab;
    private TextView jawab2;
    private TextView jawab3;

```

```

private TextView jawab4;
private TextView jawab5;
private TextView txtisi;
private TextView txtisi2;
private TextView txtisi3;
private TextView txtisi4;
private TextView txtisi5;
private List<Soal> listSoal;
private Cursor cursor = null;

Button speech, close;
private DBKamus db_kamus_helper;

@Override
public void onCreate(Bundle
savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.etika);
    db_kamus_helper = new DBKamus(this);
    try {
        db_kamus_helper.openDataBase();
        db =
db_kamus_helper.getReadableDatabase();
    } catch (SQLException sqle) {
        throw sqle;
    }

    txtisi = (TextView)
findViewById(R.id.isi1);
    txtisi2 = (TextView)
findViewById(R.id.isi2);
    txtisi3 = (TextView)
findViewById(R.id.isi3);
    txtisi4 = (TextView)
findViewById(R.id.isi4);
    txtisi5 = (TextView)
findViewById(R.id.isi5);

```

```

        jawab= (TextView)
findViewById(R.id.jawab1);
        jawab2= (TextView)
findViewById(R.id.jawab2);
        jawab3= (TextView)
findViewById(R.id.jawab3);
        jawab4= (TextView)
findViewById(R.id.jawab4);
        jawab5= (TextView)
findViewById(R.id.jawab5);

        judul = (TextView)
findViewById(R.id.title);

        close=(Button)
findViewById(R.id.close);

        listSoal = new ArrayList<Soal>();
        listSoal = getSoal();

        close.setOnClickListener(new
View.OnClickListener() {
            @Override
            public void onClick(View arg0) {
                // TODO Auto-generated method
stub
                EtikaPasar.this.finish();
            }
        });

    }

    public List<Soal> getSoal() {
        List<Soal> listSoal = new
ArrayList<Soal>();
        String query = "select * from etika";

```

```

        db =
db_kamus_helper.getReadableDatabase();
        cursor = db.rawQuery(query, null);

        if(cursor.moveToFirst()){
            do{
                Soal s = new Soal();

txtisi.setText(cursor.getString(3));

txtisi2.setText(cursor.getString(4));

txtisi3.setText(cursor.getString(5));

txtisi4.setText(cursor.getString(6));

txtisi5.setText(cursor.getString(7));


jawab.setText(cursor.getString(8));

jawab2.setText(cursor.getString(9));

jawab3.setText(cursor.getString(10));

jawab4.setText(cursor.getString(11));

jawab5.setText(cursor.getString(12));


judul.setText(cursor.getString(2));


//pic.loadUrl(cursor.getString(9));

//pic.loadUrl("file:///android_asset/img/diale
k.png");

                listSoal.add(s);
            }while(cursor.moveToNext());
        }

```

```

        return listSoal;
    }

    public boolean onCreateOptionsMenu(Menu
menu) {
        MenuInflater inflater =
getMenuInflater();

inflater.inflate(R.drawable.readermenu, menu);
        return true;
    }

    public boolean
onOptionsItemSelected(MenuItem item) {
        float current_size =
txtisi.getTextSize(); txtisi2.getTextSize();

        switch (item.getItemId()) {
            case R.id.zoomout:

txtisi.setTextSize(TypedValue.COMPLEX_UNIT_PX,
current_size*(float) (1/1.2));

txtisi2.setTextSize(TypedValue.COMPLEX_UNIT_PX
, current_size*(float) (1/1.2));

txtisi3.setTextSize(TypedValue.COMPLEX_UNIT_PX
, current_size*(float) (1/1.2));

txtisi4.setTextSize(TypedValue.COMPLEX_UNIT_PX
, current_size*(float) (1/1.2));

txtisi5.setTextSize(TypedValue.COMPLEX_UNIT_PX
, current_size*(float) (1/1.2));

jawab.setTextSize(TypedValue.COMPLEX_UNIT_PX,
current_size*(float) (1/1.2));

jawab2.setTextSize(TypedValue.COMPLEX_UNIT_PX,
current_size*(float) (1/1.2));

```



```

jawab3.setTextSize(TypedValue.COMPLEX_UNIT_PX,
current_size*(float)(1/1.2));

jawab4.setTextSize(TypedValue.COMPLEX_UNIT_PX,
current_size*(float)(1/1.2));

jawab5.setTextSize(TypedValue.COMPLEX_UNIT_PX,
current_size*(float)(1/1.2));
    break;
    case R.id.zoomin:

txtisi.setTextSize(TypedValue.COMPLEX_UNIT_PX,
current_size*(float)1.2);

txtisi2.setTextSize(TypedValue.COMPLEX_UNIT_PX
, current_size*(float)1.2);

txtisi3.setTextSize(TypedValue.COMPLEX_UNIT_PX
, current_size*(float)1.2);

txtisi4.setTextSize(TypedValue.COMPLEX_UNIT_PX
, current_size*(float)1.2);

txtisi5.setTextSize(TypedValue.COMPLEX_UNIT_PX
, current_size*(float)1.2);

jawab.setTextSize(TypedValue.COMPLEX_UNIT_PX,
current_size*(float)1.2);

jawab2.setTextSize(TypedValue.COMPLEX_UNIT_PX,
current_size*(float)1.2);

jawab3.setTextSize(TypedValue.COMPLEX_UNIT_PX,
current_size*(float)1.2);

jawab4.setTextSize(TypedValue.COMPLEX_UNIT_PX,
current_size*(float)1.2);

jawab5.setTextSize(TypedValue.COMPLEX_UNIT_PX,
current_size*(float)1.2);

```

```

        break;
        default:
            return
super.onOptionsItemSelected(item);
    }
    return true;
}
}

```

17.) Potongan kode *class* KamusDialek.

```

public class KamusDialek extends Activity
implements MediaPlayer.OnCompletionListener{

    private SQLiteDatabase db = null;
    private TextView judul;
    private TextView txtisi;
    private List<Soal> listSoal;
    private Cursor cursor = null;
    private Button play;
    private Button stop;
    private Button pause;
    private MediaPlayer mp;

    Button speech, close;
    private DBKamus db_kamus_helper;

    @Override
    public void onCreate(Bundle
savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.kamusjawa);
        db_kamus_helper = new DBKamus(this);
        try {
            db_kamus_helper.openDataBase();
            db =
db_kamus_helper.getReadableDatabase();
        } catch (SQLException sqle) {
            throw sqle;

```

```

    }

    txtisi = (TextView)
    findViewById(R.id.content);
    judul = (TextView)
    findViewById(R.id.title);

    close=(Button)
    findViewById(R.id.close);
    play=(Button) findViewById(R.id.play);

    pause=(Button) findViewById(R.id.pause);
    stop=(Button) findViewById(R.id.stop);

    listSoal = new ArrayList<Soal>();
    listSoal = getSoal();

    close.setOnClickListener(new
    View.OnClickListener() {
        @Override
        public void onClick(View arg0) {
            // TODO Auto-generated method
stub

            KamusDialek.this.finish();

        }
    });

    play.setOnClickListener(new
    View.OnClickListener() {
        public void onClick(View view) {
            play();
        }
    });

```

```

        pause.setOnClickListener(new
View.OnClickListener() {
            public void onClick(View view) {
                pause();
            }
        });

        stop.setOnClickListener(new
View.OnClickListener() {
            public void onClick(View view) {
                stop();
            }
        });

        setup();

    }

    public List<Soal> getSoal() {
        List<Soal> listSoal = new
ArrayList<Soal>();
        String query = "select * from
kamusjawa";

        db =
db_kamus_helper.getReadableDatabase();
        cursor = db.rawQuery(query, null);

        if(cursor.moveToFirst()) {
            do{
                Soal s = new Soal();

txtisi.setText(cursor.getString(2));

judul.setText(cursor.getString(1));

```

```

//pic.loadUrl(cursor.getString(9));

//pic.loadUrl("file:///android_asset/img/diale
k.png");

        listSoal.add(s);
    }while(cursor.moveToNext());
    }

    return listSoal;
}

public boolean onCreateOptionsMenu(Menu
menu) {
    MenuInflater inflater =
getMenuInflater();

inflater.inflate(R.drawable.readermenu, menu);
    return true;
}

public boolean
onOptionsItemSelected(MenuItem item) {
    float current_size =
txtisi.getTextSize();

    switch (item.getItemId()) {
        case R.id.zoomout:

txtisi.setTextSize(TypedValue.COMPLEX_UNIT_PX,
current_size*(float) (1/1.2));
        break;
        case R.id.zoomin:

txtisi.setTextSize(TypedValue.COMPLEX_UNIT_PX,
current_size*(float) 1.2);
        break;
        default:
            return
super.onOptionsItemSelected(item);

```

```

        }
        return true;
    }
    public void onDestroy() {
        super.onDestroy();

        if (stop.isEnabled()) {
            stop();
        }

        public void onCompletion(MediaPlayer mp) {
            stop();
        }

        private void play() {
            mp.start();

            play.setEnabled(false);
            pause.setEnabled(true);
            stop.setEnabled(true);
        }

        private void stop() {
            mp.stop();
            pause.setEnabled(false);
            stop.setEnabled(false);

            try {
                mp.prepare();
                mp.seekTo(0);
                play.setEnabled(true);
            }
            catch (Throwable t) {
                goBlooeY(t);
            }
        }

        private void pause() {
            mp.pause();
        }
    }

```

```

        play.setEnabled(true);
        pause.setEnabled(false);
        stop.setEnabled(true);
    }

    private void loadClip() {
        try {
            mp=MediaPlayer.create(this,
R.raw.test);
            mp.setOnCompletionListener(this);

        }
        catch (Throwable t) {
            goBloody(t);
        }
    }

    private void setup() {
        loadClip();
        play.setEnabled(true);
        pause.setEnabled(false);
        stop.setEnabled(false);

    }

    private void goBloody(Throwable t) {
        AlertDialog.Builder builder=new
AlertDialog.Builder(this);

        builder
            .setTitle("Exception!")
            .setMessage(t.toString())
            .setPositiveButton("OK", null)
            .show();
    }
}

```

18.) Potongan kode menampilkan informasi dialek pasar antara pedagang dan pembeli, contohnya salah satu contohnya pedagang ikan yakni *class* DialekIkan.

```
public class DialekIkan extends Activity
implements MediaPlayer.OnCompletionListener{

    private SQLiteDatabase db = null;
    private TextView judul;
    private TextView jawab;
    private TextView jawab2;
    private TextView jawab3;
    private TextView jawab4;
    private TextView jawab5;
    private TextView jawab6;
    private TextView jawab7;
    private TextView jawab8;
    private TextView jawab9;
    private TextView jawab10;
    private TextView jawab11;
    private TextView jawab12;
    private TextView txtisi;
    private TextView txtisi2;
    private TextView txtisi3;
    private TextView txtisi4;
    private TextView txtisi5;
    private TextView txtisi6;
    private TextView txtisi7;
    private TextView txtisi8;
    private TextView txtisi9;
    private TextView txtisi10;
    private TextView txtisi11;
    private TextView txtisi12;
    private List<Soal> listSoal;
    private Cursor cursor = null;
    private Button play;
    private Button stop;
    private Button play2;
    private Button stop2;
    private Button play3;
    private Button stop3;
    private Button play4;
```



```

private Button stop4;
private Button play5;
private Button stop5;
private Button play6;
private Button stop6;
private Button play7;
private Button stop7;
private Button play8;
private Button stop8;
private Button play9;
private Button stop9;
private Button play10;
private Button stop10;
private Button play11;
private Button stop11;
private Button play12;
private Button stop12;
private MediaPlayer mp;
private MediaPlayer media;
private MediaPlayer media3;
private MediaPlayer media4;
private MediaPlayer media5;
private MediaPlayer media6;
private MediaPlayer media7;
private MediaPlayer media8;
private MediaPlayer media9;
private MediaPlayer media10;
private MediaPlayer media11;
private MediaPlayer media12;

Button speech, close;
private DBKamus db_kamus_helper;

@Override
public void onCreate(Bundle
savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.bandeng);
    db_kamus_helper = new DBKamus(this);

```

```
        try {
            db_kamus_helper.openDataBase();
            db =
db_kamus_helper.getReadableDatabase();
        } catch (SQLException sqle) {
            throw sqle;
        }

        txtisi = (TextView)
findViewById(R.id.isi1);
        txtisi2 = (TextView)
findViewById(R.id.isi2);
        txtisi3 = (TextView)
findViewById(R.id.isi3);
        txtisi4 = (TextView)
findViewById(R.id.isi4);
        txtisi5 = (TextView)
findViewById(R.id.isi5);
        txtisi6 = (TextView)
findViewById(R.id.isi6);
        txtisi7 = (TextView)
findViewById(R.id.isi7);
        txtisi8 = (TextView)
findViewById(R.id.isi8);
        txtisi9 = (TextView)
findViewById(R.id.isi9);
        txtisi10 = (TextView)
findViewById(R.id.isi10);
        txtisi11= (TextView)
findViewById(R.id.isi11);
        txtisi12= (TextView)
findViewById(R.id.isi12);
        jawab= (TextView)
findViewById(R.id.jawab1);
        jawab2= (TextView)
findViewById(R.id.jawab2);
        jawab3= (TextView)
findViewById(R.id.jawab3);
        jawab4= (TextView)
findViewById(R.id.jawab4);
```

```
        jawab5= (TextView)
findViewById(R.id.jawab5);
        jawab6= (TextView)
findViewById(R.id.jawab6);
        jawab7= (TextView)
findViewById(R.id.jawab7);
        jawab8= (TextView)
findViewById(R.id.jawab8);
        jawab9= (TextView)
findViewById(R.id.jawab9);
        jawab10= (TextView)
findViewById(R.id.jawab10);
        jawab11= (TextView)
findViewById(R.id.jawab11);
        jawab12= (TextView)
findViewById(R.id.jawab12);
        judul = (TextView)
findViewById(R.id.title);

        close=(Button)
findViewById(R.id.close);
        play=(Button) findViewById(R.id.play);
        stop=(Button) findViewById(R.id.stop);

play2=(Button) findViewById(R.id.play2);

stop2=(Button) findViewById(R.id.stop2);

play3=(Button) findViewById(R.id.play3);

stop3=(Button) findViewById(R.id.stop3);

play4=(Button) findViewById(R.id.play4);

stop4=(Button) findViewById(R.id.stop4);

play5=(Button) findViewById(R.id.play5);

stop5=(Button) findViewById(R.id.stop5);
```

```

play6=(Button) findViewById(R.id.play6);
stop6=(Button) findViewById(R.id.stop6);
play7=(Button) findViewById(R.id.play7);
stop7=(Button) findViewById(R.id.stop7);
play8=(Button) findViewById(R.id.play8);
stop8=(Button) findViewById(R.id.stop8);
play9=(Button) findViewById(R.id.play9);
stop9=(Button) findViewById(R.id.stop9);
play10=(Button) findViewById(R.id.play10);
stop10=(Button) findViewById(R.id.stop10);
play11=(Button) findViewById(R.id.play11);
stop11=(Button) findViewById(R.id.stop11);
play12=(Button) findViewById(R.id.play12);
stop12=(Button) findViewById(R.id.stop12);

        listSoal = new ArrayList<Soal>();
        listSoal = getSoal();

        close.setOnClickListener(new
View.OnClickListener() {
            @Override
            public void onClick(View arg0) {
                // TODO Auto-generated method
stub

                DialekIkan.this.finish();

```

```

        }
    });

    play.setOnClickListener(new
View.OnClickListener() {
        public void onClick(View view) {
            play();
        }
    });

    stop.setOnClickListener(new
View.OnClickListener() {
        public void onClick(View view) {
            stop();
        }
    });

    play2.setOnClickListener(new
View.OnClickListener() {
        public void onClick(View view) {
            play2();
        }
    });

    stop2.setOnClickListener(new
View.OnClickListener() {
        public void onClick(View view) {
            stop2();
        }
    });

    play3.setOnClickListener(new
View.OnClickListener() {
        public void onClick(View view) {
            play3();
        }
    });

    stop3.setOnClickListener(new
View.OnClickListener() {

```

```
        public void onClick(View view) {
            stop3();
        }
    });

    play4.setOnClickListener(new
View.OnClickListener() {
        public void onClick(View view) {
            play4();
        }
    });

    stop4.setOnClickListener(new
View.OnClickListener() {
        public void onClick(View view) {
            stop4();
        }
    });

    play5.setOnClickListener(new
View.OnClickListener() {
        public void onClick(View view) {
            play5();
        }
    });

    stop5.setOnClickListener(new
View.OnClickListener() {
        public void onClick(View view) {
            stop5();
        }
    });

    play6.setOnClickListener(new
View.OnClickListener() {
        public void onClick(View view) {
            play6();
        }
    });

    stop6.setOnClickListener(new
View.OnClickListener() {
```

```
        public void onClick(View view) {
            stop6();
        }
    });

    play7.setOnClickListener(new
View.OnClickListener() {
        public void onClick(View view) {
            play7();
        }
    });

    stop7.setOnClickListener(new
View.OnClickListener() {
        public void onClick(View view) {
            stop7();
        }
    });

    play8.setOnClickListener(new
View.OnClickListener() {
        public void onClick(View view) {
            play8();
        }
    });

    stop8.setOnClickListener(new
View.OnClickListener() {
        public void onClick(View view) {
            stop8();
        }
    });

    play9.setOnClickListener(new
View.OnClickListener() {
        public void onClick(View view) {
            play9();
        }
    });
});
```

```
        stop9.setOnClickListener(new
View.OnClickListener() {
            public void onClick(View view) {
                stop9();
            }
        });

        play10.setOnClickListener(new
View.OnClickListener() {
            public void onClick(View view) {
                play10();
            }
        });

        stop10.setOnClickListener(new
View.OnClickListener() {
            public void onClick(View view) {
                stop10();
            }
        });

        play11.setOnClickListener(new
View.OnClickListener() {
            public void onClick(View view) {
                play11();
            }
        });

        stop11.setOnClickListener(new
View.OnClickListener() {
            public void onClick(View view) {
                stop11();
            }
        });

        play12.setOnClickListener(new
View.OnClickListener() {
            public void onClick(View view) {
                play12();
            }
        });
    });
}
```



```

        stop12.setOnClickListener(new
View.OnClickListener() {
            public void onClick(View view) {
                stop12();
            }
        });

        setup();

    }

    public List<Soal> getSoal(){
        List<Soal> listSoal = new
ArrayList<Soal>();
        String query = "select * from jawaku";

        db =
db_kamus_helper.getReadableDatabase();
        cursor = db.rawQuery(query, null);

        if(cursor.moveToFirst()){
            do{
                Soal s = new Soal();

txtisi.setText(cursor.getString(3));

txtisi2.setText(cursor.getString(4));

txtisi3.setText(cursor.getString(5));

txtisi4.setText(cursor.getString(6));

txtisi5.setText(cursor.getString(7));

txtisi6.setText(cursor.getString(8));

txtisi7.setText(cursor.getString(9));

txtisi8.setText(cursor.getString(10));

```

H62

```
txtisi9.setText(cursor.getString(11));
txtisi10.setText(cursor.getString(12));
txtisi11.setText(cursor.getString(13));
txtisi12.setText(cursor.getString(14));
jawab.setText(cursor.getString(15));
jawab2.setText(cursor.getString(16));
jawab3.setText(cursor.getString(17));
jawab4.setText(cursor.getString(18));
jawab5.setText(cursor.getString(19));
jawab6.setText(cursor.getString(20));
jawab7.setText(cursor.getString(21));
jawab8.setText(cursor.getString(22));
jawab9.setText(cursor.getString(23));
jawab10.setText(cursor.getString(24));
jawab11.setText(cursor.getString(25));
jawab12.setText(cursor.getString(26));
judul.setText(cursor.getString(2));

//pic.loadUrl(cursor.getString(9));

//pic.loadUrl("file:///android_asset/img/diale
k.png");
listSoal.add(s);
```

```

        }while(cursor.moveToNext());
    }

    return listSoal;
}

public boolean onCreateOptionsMenu(Menu
menu) {
    MenuInflater inflater =
getMenuInflater();

inflater.inflate(R.drawable.readermenu, menu);
    return true;
}

public boolean
onOptionsItemSelected(MenuItem item) {
    float current_size =
txtisi.getTextSize(); txtisi2.getTextSize();

    switch (item.getItemId()) {
        case R.id.zoomout:

txtisi.setTextSize(TypedValue.COMPLEX_UNIT_PX,
current_size*(float) (1/1.2));

txtisi2.setTextSize(TypedValue.COMPLEX_UNIT_PX
, current_size*(float) (1/1.2));

txtisi3.setTextSize(TypedValue.COMPLEX_UNIT_PX
, current_size*(float) (1/1.2));

txtisi4.setTextSize(TypedValue.COMPLEX_UNIT_PX
, current_size*(float) (1/1.2));

txtisi5.setTextSize(TypedValue.COMPLEX_UNIT_PX
, current_size*(float) (1/1.2));

txtisi6.setTextSize(TypedValue.COMPLEX_UNIT_PX
, current_size*(float) (1/1.2));

```

```
txtisi7.setTextSize(TypedValue.COMPLEX_UNIT_PX
, current_size*(float) (1/1.2));

txtisi8.setTextSize(TypedValue.COMPLEX_UNIT_PX
, current_size*(float) (1/1.2));

txtisi9.setTextSize(TypedValue.COMPLEX_UNIT_PX
, current_size*(float) (1/1.2));

txtisi10.setTextSize(TypedValue.COMPLEX_UNIT_P
X, current_size*(float) (1/1.2));

txtisi11.setTextSize(TypedValue.COMPLEX_UNIT_P
X, current_size*(float) (1/1.2));

txtisi12.setTextSize(TypedValue.COMPLEX_UNIT_P
X, current_size*(float) (1/1.2));

jawab.setTextSize(TypedValue.COMPLEX_UNIT_PX,
current_size*(float) (1/1.2));

jawab2.setTextSize(TypedValue.COMPLEX_UNIT_PX,
current_size*(float) (1/1.2));

jawab3.setTextSize(TypedValue.COMPLEX_UNIT_PX,
current_size*(float) (1/1.2));

jawab4.setTextSize(TypedValue.COMPLEX_UNIT_PX,
current_size*(float) (1/1.2));

jawab5.setTextSize(TypedValue.COMPLEX_UNIT_PX,
current_size*(float) (1/1.2));

jawab6.setTextSize(TypedValue.COMPLEX_UNIT_PX,
current_size*(float) (1/1.2));

jawab7.setTextSize(TypedValue.COMPLEX_UNIT_PX,
current_size*(float) (1/1.2));

jawab8.setTextSize(TypedValue.COMPLEX_UNIT_PX,
current_size*(float) (1/1.2));
```

```

jawab9.setTextSize(TypedValue.COMPLEX_UNIT_PX,
current_size*(float)(1/1.2));

jawab10.setTextSize(TypedValue.COMPLEX_UNIT_PX
, current_size*(float)(1/1.2));

jawab11.setTextSize(TypedValue.COMPLEX_UNIT_PX
, current_size*(float)(1/1.2));

jawab12.setTextSize(TypedValue.COMPLEX_UNIT_PX
, current_size*(float)(1/1.2));
    break;
    case R.id.zoomin:

txtisi.setTextSize(TypedValue.COMPLEX_UNIT_PX,
current_size*(float)1.2);

txtisi2.setTextSize(TypedValue.COMPLEX_UNIT_PX
, current_size*(float)1.2);

txtisi3.setTextSize(TypedValue.COMPLEX_UNIT_PX
, current_size*(float)1.2);

txtisi4.setTextSize(TypedValue.COMPLEX_UNIT_PX
, current_size*(float)1.2);

txtisi5.setTextSize(TypedValue.COMPLEX_UNIT_PX
, current_size*(float)1.2);

txtisi6.setTextSize(TypedValue.COMPLEX_UNIT_PX
, current_size*(float)1.2);

txtisi7.setTextSize(TypedValue.COMPLEX_UNIT_PX
, current_size*(float)1.2);

txtisi8.setTextSize(TypedValue.COMPLEX_UNIT_PX
, current_size*(float)1.2);

txtisi9.setTextSize(TypedValue.COMPLEX_UNIT_PX
, current_size*(float)1.2);

```

```
txtisil0.setTextSize(TypedValue.COMPLEX_UNIT_PX, current_size*(float)1.2);

txtisil1.setTextSize(TypedValue.COMPLEX_UNIT_PX, current_size*(float)1.2);

txtisil2.setTextSize(TypedValue.COMPLEX_UNIT_PX, current_size*(float)1.2);

jawab.setTextSize(TypedValue.COMPLEX_UNIT_PX, current_size*(float)1.2);

jawab2.setTextSize(TypedValue.COMPLEX_UNIT_PX, current_size*(float)1.2);

jawab3.setTextSize(TypedValue.COMPLEX_UNIT_PX, current_size*(float)1.2);

jawab4.setTextSize(TypedValue.COMPLEX_UNIT_PX, current_size*(float)1.2);

jawab5.setTextSize(TypedValue.COMPLEX_UNIT_PX, current_size*(float)1.2);

jawab6.setTextSize(TypedValue.COMPLEX_UNIT_PX, current_size*(float)1.2);

jawab7.setTextSize(TypedValue.COMPLEX_UNIT_PX, current_size*(float)1.2);

jawab8.setTextSize(TypedValue.COMPLEX_UNIT_PX, current_size*(float)1.2);

jawab9.setTextSize(TypedValue.COMPLEX_UNIT_PX, current_size*(float)1.2);

jawab10.setTextSize(TypedValue.COMPLEX_UNIT_PX, current_size*(float)1.2);

jawab11.setTextSize(TypedValue.COMPLEX_UNIT_PX, current_size*(float)1.2);
```

```

jawab12.setTextSize(TypedValue.COMPLEX_UNIT_PX
, current_size*(float)1.2);
    break;
    default:
        return
super.onOptionsItemSelected(item);
    }
    return true;
}
public void onDestroy() {
    super.onDestroy();

    if (stop.isEnabled()) {
        stop();}
    if (stop2.isEnabled()) {
        stop2();}
    if (stop3.isEnabled()) {
        stop3();}
    if (stop4.isEnabled()) {
        stop4();}
    if (stop5.isEnabled()) {
        stop5();}
    if (stop6.isEnabled()) {
        stop6();}
    if (stop7.isEnabled()) {
        stop7();}
    if (stop8.isEnabled()) {
        stop8();}
    if (stop9.isEnabled()) {
        stop9();}
    if (stop10.isEnabled()) {
        stop10();}
    if (stop11.isEnabled()) {
        stop11();}
    if (stop12.isEnabled()) {
        stop12();}

}

public void onCompletion(MediaPlayer mp) {
    stop();

```

```
        stop2();
        stop3();
        stop4();
        stop5();
        stop6();
        stop7();
        stop8();
        stop9();
        stop10();
        stop11();
        stop12();
    }

    private void play() {
        mp.start();

        play.setEnabled(false);

        stop.setEnabled(true);
    }

    private void stop() {
        mp.stop();

        stop.setEnabled(false);

        try {
            mp.prepare();
            mp.seekTo(0);
            play.setEnabled(true);
        }
        catch (Throwable t) {
            goBlooley(t);
        }
    }

    private void play2() {
```



```
        media.start();

        play2.setEnabled(false);

        stop2.setEnabled(true);
    }

    private void stop2() {
        media.stop();

        stop2.setEnabled(false);

        try {
            media.prepare();
            media.seekTo(0);
            play2.setEnabled(true);
        }
        catch (Throwable t) {
            goBlooeY(t);
        }
    }

    private void play3() {
        media3.start();

        play3.setEnabled(false);

        stop3.setEnabled(true);
    }

    private void stop3() {
        media3.stop();

        stop3.setEnabled(false);

        try {
            media3.prepare();
            media3.seekTo(0);
            play3.setEnabled(true);
```

```
    }  
    catch (Throwable t) {  
        goBlooey(t);  
    }  
}  
  
private void play4() {  
    media4.start();  
  
    play4.setEnabled(false);  
  
    stop4.setEnabled(true);  
}  
  
private void stop4() {  
    media4.stop();  
  
    stop4.setEnabled(false);  
  
    try {  
        media4.prepare();  
        media4.seekTo(0);  
        play4.setEnabled(true);  
    }  
    catch (Throwable t) {  
        goBlooey(t);  
    }  
}  
  
private void play5() {  
    media5.start();  
  
    play5.setEnabled(false);  
  
    stop5.setEnabled(true);  
}  
  
private void stop5() {  
    media5.stop();  
}
```

```
        stop5.setEnabled(false);

        try {
            media5.prepare();
            media5.seekTo(0);
            play5.setEnabled(true);
        }
        catch (Throwable t) {
            goBlooley(t);
        }
    }

    private void play6() {
        media6.start();

        play6.setEnabled(false);

        stop6.setEnabled(true);
    }

    private void stop6() {
        media6.stop();

        stop6.setEnabled(false);

        try {
            media6.prepare();
            media6.seekTo(0);
            play6.setEnabled(true);
        }
        catch (Throwable t) {
            goBlooley(t);
        }
    }

    private void play7() {
        media7.start();

        play7.setEnabled(false);
```

```
        stop7.setEnabled(true);
    }

    private void stop7() {
        media7.stop();

        stop7.setEnabled(false);

        try {
            media7.prepare();
            media7.seekTo(0);
            play7.setEnabled(true);
        }
        catch (Throwable t) {
            goBlooey(t);
        }
    }

    private void play8() {
        media8.start();

        play8.setEnabled(false);

        stop8.setEnabled(true);
    }

    private void stop8() {
        media8.stop();

        stop8.setEnabled(false);

        try {
            media8.prepare();
            media8.seekTo(0);
            play8.setEnabled(true);
        }
        catch (Throwable t) {
            goBlooey(t);
        }
    }
}
```

```
}

private void play9() {
    media9.start();

    play9.setEnabled(false);

    stop9.setEnabled(true);
}

private void stop9() {
    media9.stop();

    stop9.setEnabled(false);

    try {
        media9.prepare();
        media9.seekTo(0);
        play9.setEnabled(true);
    }
    catch (Throwable t) {
        goBlooeY(t);
    }
}

private void play10() {
    media10.start();

    play10.setEnabled(false);

    stop10.setEnabled(true);
}

private void stop10() {
    media10.stop();

    stop10.setEnabled(false);

    try {
```

```
        media10.prepare();
        media10.seekTo(0);
        play10.setEnabled(true);
    }
    catch (Throwable t) {
        goBlooeey(t);
    }
}

private void play11() {
    media11.start();

    play11.setEnabled(false);

    stop11.setEnabled(true);
}

private void stop11() {
    media11.stop();

    stop11.setEnabled(false);

    try {
        media11.prepare();
        media11.seekTo(0);
        play11.setEnabled(true);
    }
    catch (Throwable t) {
        goBlooeey(t);
    }
}

private void play12() {
    media12.start();

    play12.setEnabled(false);

    stop12.setEnabled(true);
}
```

```

private void stop12() {
    media12.stop();

    stop12.setEnabled(false);

    try {
        media12.prepare();
        media12.seekTo(0);
        play12.setEnabled(true);
    }
    catch (Throwable t) {
        goBlooney(t);
    }
}

private void loadClip() {
    try {
        mp=MediaPlayer.create(this,
R.raw.dialek);
        mp.setOnCompletionListener(this);

        media=MediaPlayer.create(this,
R.raw.dialekduapuluh);

media.setOnCompletionListener(this);

        media3=MediaPlayer.create(this,
R.raw.dialekenambelas);

media3.setOnCompletionListener(this);

        media4=MediaPlayer.create(this,
R.raw.dialekempatbelas);

media4.setOnCompletionListener(this);

        media5=MediaPlayer.create(this,
R.raw.dialekduasembilan);

media5.setOnCompletionListener(this);

```

```

        media6=MediaPlayer.create(this,
R.raw.dialektigadua);
media6.setOnCompletionListener(this);

        media7=MediaPlayer.create(this,
R.raw.dialektigaempat);
media7.setOnCompletionListener(this);

        media8=MediaPlayer.create(this,
R.raw.dialeksebelas);
media8.setOnCompletionListener(this);

        media9=MediaPlayer.create(this,
R.raw.dialektigatiga);
media9.setOnCompletionListener(this);

        media10=MediaPlayer.create(this,
R.raw.dialektigalima);
media10.setOnCompletionListener(this);

        media11=MediaPlayer.create(this,
R.raw.dialektigaenam);
media11.setOnCompletionListener(this);

        media12=MediaPlayer.create(this,
R.raw.dialektigatujuh);
media12.setOnCompletionListener(this);
    }
    catch (Throwable t) {
        goBlooeey(t);
    }
}

private void setup() {

```



```

        loadClip();
        play.setEnabled(true);
        stop.setEnabled(false);
        play2.setEnabled(true);
        stop2.setEnabled(false);
        play3.setEnabled(true);
        stop3.setEnabled(false);
        play4.setEnabled(true);
        stop4.setEnabled(false);
        play5.setEnabled(true);
        stop5.setEnabled(false);
        play6.setEnabled(true);
        stop6.setEnabled(false);
        play7.setEnabled(true);
        stop7.setEnabled(false);
        play8.setEnabled(true);
        stop8.setEnabled(false);
        play9.setEnabled(true);
        stop9.setEnabled(false);
        play10.setEnabled(true);
        stop10.setEnabled(false);
        play11.setEnabled(true);
        stop11.setEnabled(false);
        play12.setEnabled(true);
        stop12.setEnabled(false);
    }

    private void goBlooeey(Throwable t) {
        AlertDialog.Builder builder=new
        AlertDialog.Builder(this);

        builder
            .setTitle("Exception!")
            .setMessage(t.toString())
            .setPositiveButton("OK", null)
            .show();
    }
}

```

19.) Potongan kode menampilkan informasi dialek pasar antara pedagang dan pembeli, contohnya salah satu contohnya pedagang buah yakni *class DialekBuah*.

```
public class DialekBuah extends Activity
implements MediaPlayer.OnCompletionListener{

    private SQLiteDatabase db = null;
    private TextView judul;
    private TextView jawab;
    private TextView jawab2;
    private TextView jawab3;
    private TextView jawab4;
    private TextView jawab5;
    private TextView jawab6;
    private TextView jawab7;
    private TextView jawab8;
    private TextView jawab9;
    private TextView jawab10;
    private TextView jawab11;
    private TextView jawab12;
    private TextView txtisi;
    private TextView txtisi2;
    private TextView txtisi3;
    private TextView txtisi4;
    private TextView txtisi5;
    private TextView txtisi6;
    private TextView txtisi7;
    private TextView txtisi8;
    private TextView txtisi9;
    private TextView txtisi10;
    private TextView txtisi11;
    private TextView txtisi12;
    private List<Soal> listSoal;
    private Cursor cursor = null;
    private Button play;
    private Button stop;
    private Button play2;
    private Button stop2;
    private Button play3;
    private Button stop3;
    private Button play4;
```

```

private Button stop4;
private Button play5;
private Button stop5;
private Button play6;
private Button stop6;
private Button play7;
private Button stop7;
private Button play8;
private Button stop8;
private Button play9;
private Button stop9;
private Button play10;
private Button stop10;
private Button play11;
private Button stop11;
private Button play12;
private Button stop12;
private MediaPlayer mp;
private MediaPlayer media;
private MediaPlayer media3;
private MediaPlayer media4;
private MediaPlayer media5;
private MediaPlayer media6;
private MediaPlayer media7;
private MediaPlayer media8;
private MediaPlayer media9;
private MediaPlayer media10;
private MediaPlayer media11;
private MediaPlayer media12;

Button speech, close;
private DBKamus db_kamus_helper;

@Override
public void onCreate(Bundle
savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.bandeng);
    db_kamus_helper = new DBKamus(this);

```

```
        try {
            db_kamus_helper.openDataBase();
            db =
db_kamus_helper.getReadableDatabase();
        } catch (SQLException sqle) {
            throw sqle;
        }

        txtisi = (TextView)
findViewById(R.id.isi1);
        txtisi2 = (TextView)
findViewById(R.id.isi2);
        txtisi3 = (TextView)
findViewById(R.id.isi3);
        txtisi4 = (TextView)
findViewById(R.id.isi4);
        txtisi5 = (TextView)
findViewById(R.id.isi5);
        txtisi6 = (TextView)
findViewById(R.id.isi6);
        txtisi7 = (TextView)
findViewById(R.id.isi7);
        txtisi8 = (TextView)
findViewById(R.id.isi8);
        txtisi9 = (TextView)
findViewById(R.id.isi9);
        txtisi10 = (TextView)
findViewById(R.id.isi10);
        txtisi11= (TextView)
findViewById(R.id.isi11);
        txtisi12= (TextView)
findViewById(R.id.isi12);
        jawab= (TextView)
findViewById(R.id.jawab1);
        jawab2= (TextView)
findViewById(R.id.jawab2);
        jawab3= (TextView)
findViewById(R.id.jawab3);
        jawab4= (TextView)
findViewById(R.id.jawab4);
```

```
        jawab5= (TextView)
findViewById(R.id.jawab5);
        jawab6= (TextView)
findViewById(R.id.jawab6);
        jawab7= (TextView)
findViewById(R.id.jawab7);
        jawab8= (TextView)
findViewById(R.id.jawab8);
        jawab9= (TextView)
findViewById(R.id.jawab9);
        jawab10= (TextView)
findViewById(R.id.jawab10);
        jawab11= (TextView)
findViewById(R.id.jawab11);
        jawab12= (TextView)
findViewById(R.id.jawab12);
        judul = (TextView)
findViewById(R.id.title);

        close=(Button)
findViewById(R.id.close);
        play=(Button) findViewById(R.id.play);
        stop=(Button) findViewById(R.id.stop);

play2=(Button) findViewById(R.id.play2);

stop2=(Button) findViewById(R.id.stop2);

play3=(Button) findViewById(R.id.play3);

stop3=(Button) findViewById(R.id.stop3);

play4=(Button) findViewById(R.id.play4);

stop4=(Button) findViewById(R.id.stop4);

play5=(Button) findViewById(R.id.play5);

stop5=(Button) findViewById(R.id.stop5);
```

```
play6=(Button) findViewById(R.id.play6);
stop6=(Button) findViewById(R.id.stop6);
play7=(Button) findViewById(R.id.play7);
stop7=(Button) findViewById(R.id.stop7);
play8=(Button) findViewById(R.id.play8);
stop8=(Button) findViewById(R.id.stop8);
play9=(Button) findViewById(R.id.play9);
stop9=(Button) findViewById(R.id.stop9);
play10=(Button) findViewById(R.id.play10);
stop10=(Button) findViewById(R.id.stop10);
play11=(Button) findViewById(R.id.play11);
stop11=(Button) findViewById(R.id.stop11);
play12=(Button) findViewById(R.id.play12);
stop12=(Button) findViewById(R.id.stop12);

        listSoal = new ArrayList<Soal>();
        listSoal = getSoal();

        close.setOnClickListener(new
View.OnClickListener() {
            @Override
            public void onClick(View arg0) {
                // TODO Auto-generated method
stub

                DialekBuah.this.finish();
```

```

        }
    });

    play.setOnClickListener(new
View.OnClickListener() {
        public void onClick(View view) {
            play();
        }
    });

    stop.setOnClickListener(new
View.OnClickListener() {
        public void onClick(View view) {
            stop();
        }
    });

    play2.setOnClickListener(new
View.OnClickListener() {
        public void onClick(View view) {
            play2();
        }
    });

    stop2.setOnClickListener(new
View.OnClickListener() {
        public void onClick(View view) {
            stop2();
        }
    });

    play3.setOnClickListener(new
View.OnClickListener() {
        public void onClick(View view) {
            play3();
        }
    });

    stop3.setOnClickListener(new
View.OnClickListener() {

```

```
        public void onClick(View view) {
            stop3();
        }
    });

    play4.setOnClickListener(new
View.OnClickListener() {
        public void onClick(View view) {
            play4();
        }
    });

    stop4.setOnClickListener(new
View.OnClickListener() {
        public void onClick(View view) {
            stop4();
        }
    });

    play5.setOnClickListener(new
View.OnClickListener() {
        public void onClick(View view) {
            play5();
        }
    });

    stop5.setOnClickListener(new
View.OnClickListener() {
        public void onClick(View view) {
            stop5();
        }
    });

    play6.setOnClickListener(new
View.OnClickListener() {
        public void onClick(View view) {
            play6();
        }
    });

    stop6.setOnClickListener(new
View.OnClickListener() {
```



```
        public void onClick(View view) {
            stop6();
        }
    });

    play7.setOnClickListener(new
View.OnClickListener() {
        public void onClick(View view) {
            play7();
        }
    });

    stop7.setOnClickListener(new
View.OnClickListener() {
        public void onClick(View view) {
            stop7();
        }
    });

    play8.setOnClickListener(new
View.OnClickListener() {
        public void onClick(View view) {
            play8();
        }
    });

    stop8.setOnClickListener(new
View.OnClickListener() {
        public void onClick(View view) {
            stop8();
        }
    });

    play9.setOnClickListener(new
View.OnClickListener() {
        public void onClick(View view) {
            play9();
        }
    });
});
```

```
        stop9.setOnClickListener(new
View.OnClickListener() {
            public void onClick(View view) {
                stop9();
            }
        });

        play10.setOnClickListener(new
View.OnClickListener() {
            public void onClick(View view) {
                play10();
            }
        });

        stop10.setOnClickListener(new
View.OnClickListener() {
            public void onClick(View view) {
                stop10();
            }
        });

        play11.setOnClickListener(new
View.OnClickListener() {
            public void onClick(View view) {
                play11();
            }
        });

        stop11.setOnClickListener(new
View.OnClickListener() {
            public void onClick(View view) {
                stop11();
            }
        });

        play12.setOnClickListener(new
View.OnClickListener() {
            public void onClick(View view) {
                play12();
            }
        });
    });
}
```

```

        stop12.setOnClickListener(new
View.OnClickListener() {
            public void onClick(View view) {
                stop12();
            }
        });

        setup();

    }

    public List<Soal> getSoal(){
        List<Soal> listSoal = new
ArrayList<Soal>();
        String query = "select * from
dialekbuah";

        db =
db_kamus_helper.getReadableDatabase();
        cursor = db.rawQuery(query, null);

        if(cursor.moveToFirst()){
            do{
                Soal s = new Soal();

txtisi.setText(cursor.getString(3));

txtisi2.setText(cursor.getString(4));

txtisi3.setText(cursor.getString(5));

txtisi4.setText(cursor.getString(6));

txtisi5.setText(cursor.getString(7));

txtisi6.setText(cursor.getString(8));

txtisi7.setText(cursor.getString(9));

```

```
txtisi8.setText(cursor.getString(10));  
txtisi9.setText(cursor.getString(11));  
txtisi10.setText(cursor.getString(12));  
txtisi11.setText(cursor.getString(13));  
txtisi12.setText(cursor.getString(14));  
jawab.setText(cursor.getString(15));  
jawab2.setText(cursor.getString(16));  
jawab3.setText(cursor.getString(17));  
jawab4.setText(cursor.getString(18));  
jawab5.setText(cursor.getString(19));  
jawab6.setText(cursor.getString(20));  
jawab7.setText(cursor.getString(21));  
jawab8.setText(cursor.getString(22));  
jawab9.setText(cursor.getString(23));  
jawab10.setText(cursor.getString(24));  
jawab11.setText(cursor.getString(25));  
jawab12.setText(cursor.getString(26));  
judul.setText(cursor.getString(2));  
  
//pic.loadUrl(cursor.getString(9));
```

```

//pic.loadUrl("file:///android_asset/img/diale
k.png");

        listSoal.add(s);
    }while(cursor.moveToNext());
    }

    return listSoal;
}

    public boolean onCreateOptionsMenu(Menu
menu) {
        MenuInflater inflater =
getMenuInflater();

inflater.inflate(R.drawable.readermenu, menu);
        return true;
    }

    public boolean
onOptionsItemSelected(MenuItem item) {
        float current_size =
txtisi.getTextSize(); txtisi2.getTextSize();

        switch (item.getItemId()) {
            case R.id.zoomout:

txtisi.setTextSize(TypedValue.COMPLEX_UNIT_PX,
current_size*(float) (1/1.2));

txtisi2.setTextSize(TypedValue.COMPLEX_UNIT_PX
, current_size*(float) (1/1.2));

txtisi3.setTextSize(TypedValue.COMPLEX_UNIT_PX
, current_size*(float) (1/1.2));

txtisi4.setTextSize(TypedValue.COMPLEX_UNIT_PX
, current_size*(float) (1/1.2));

txtisi5.setTextSize(TypedValue.COMPLEX_UNIT_PX
, current_size*(float) (1/1.2));

```

```
txtisi6.setTextSize(TypedValue.COMPLEX_UNIT_PX
, current_size*(float) (1/1.2));

txtisi7.setTextSize(TypedValue.COMPLEX_UNIT_PX
, current_size*(float) (1/1.2));

txtisi8.setTextSize(TypedValue.COMPLEX_UNIT_PX
, current_size*(float) (1/1.2));

txtisi9.setTextSize(TypedValue.COMPLEX_UNIT_PX
, current_size*(float) (1/1.2));

txtisi10.setTextSize(TypedValue.COMPLEX_UNIT_P
X, current_size*(float) (1/1.2));

txtisi11.setTextSize(TypedValue.COMPLEX_UNIT_P
X, current_size*(float) (1/1.2));

txtisi12.setTextSize(TypedValue.COMPLEX_UNIT_P
X, current_size*(float) (1/1.2));

jawab.setTextSize(TypedValue.COMPLEX_UNIT_PX,
current_size*(float) (1/1.2));

jawab2.setTextSize(TypedValue.COMPLEX_UNIT_PX,
current_size*(float) (1/1.2));

jawab3.setTextSize(TypedValue.COMPLEX_UNIT_PX,
current_size*(float) (1/1.2));

jawab4.setTextSize(TypedValue.COMPLEX_UNIT_PX,
current_size*(float) (1/1.2));

jawab5.setTextSize(TypedValue.COMPLEX_UNIT_PX,
current_size*(float) (1/1.2));

jawab6.setTextSize(TypedValue.COMPLEX_UNIT_PX,
current_size*(float) (1/1.2));

jawab7.setTextSize(TypedValue.COMPLEX_UNIT_PX,
current_size*(float) (1/1.2));
```

```

jawab8.setTextSize(TypedValue.COMPLEX_UNIT_PX,
current_size*(float)(1/1.2));

jawab9.setTextSize(TypedValue.COMPLEX_UNIT_PX,
current_size*(float)(1/1.2));

jawab10.setTextSize(TypedValue.COMPLEX_UNIT_PX
, current_size*(float)(1/1.2));

jawab11.setTextSize(TypedValue.COMPLEX_UNIT_PX
, current_size*(float)(1/1.2));

jawab12.setTextSize(TypedValue.COMPLEX_UNIT_PX
, current_size*(float)(1/1.2));
    break;
    case R.id.zoomin:

txtisi.setTextSize(TypedValue.COMPLEX_UNIT_PX,
current_size*(float)1.2);

txtisi2.setTextSize(TypedValue.COMPLEX_UNIT_PX
, current_size*(float)1.2);

txtisi3.setTextSize(TypedValue.COMPLEX_UNIT_PX
, current_size*(float)1.2);

txtisi4.setTextSize(TypedValue.COMPLEX_UNIT_PX
, current_size*(float)1.2);

txtisi5.setTextSize(TypedValue.COMPLEX_UNIT_PX
, current_size*(float)1.2);

txtisi6.setTextSize(TypedValue.COMPLEX_UNIT_PX
, current_size*(float)1.2);

txtisi7.setTextSize(TypedValue.COMPLEX_UNIT_PX
, current_size*(float)1.2);

txtisi8.setTextSize(TypedValue.COMPLEX_UNIT_PX
, current_size*(float)1.2);

```

```
txtisi9.setTextSize(TypedValue.COMPLEX_UNIT_PX
, current_size*(float)1.2);

txtisi10.setTextSize(TypedValue.COMPLEX_UNIT_P
X, current_size*(float)1.2);

txtisi11.setTextSize(TypedValue.COMPLEX_UNIT_P
X, current_size*(float)1.2);

txtisi12.setTextSize(TypedValue.COMPLEX_UNIT_P
X, current_size*(float)1.2);

jawab.setTextSize(TypedValue.COMPLEX_UNIT_PX,
current_size*(float)1.2);

jawab2.setTextSize(TypedValue.COMPLEX_UNIT_PX,
current_size*(float)1.2);

jawab3.setTextSize(TypedValue.COMPLEX_UNIT_PX,
current_size*(float)1.2);

jawab4.setTextSize(TypedValue.COMPLEX_UNIT_PX,
current_size*(float)1.2);

jawab5.setTextSize(TypedValue.COMPLEX_UNIT_PX,
current_size*(float)1.2);

jawab6.setTextSize(TypedValue.COMPLEX_UNIT_PX,
current_size*(float)1.2);

jawab7.setTextSize(TypedValue.COMPLEX_UNIT_PX,
current_size*(float)1.2);

jawab8.setTextSize(TypedValue.COMPLEX_UNIT_PX,
current_size*(float)1.2);

jawab9.setTextSize(TypedValue.COMPLEX_UNIT_PX,
current_size*(float)1.2);

jawab10.setTextSize(TypedValue.COMPLEX_UNIT_PX
, current_size*(float)1.2);
```



```

jawab11.setTextSize(TypedValue.COMPLEX_UNIT_PX
, current_size*(float)1.2);

jawab12.setTextSize(TypedValue.COMPLEX_UNIT_PX
, current_size*(float)1.2);
    break;
    default:
        return
super.onOptionsItemSelected(item);
    }
    return true;
}
public void onDestroy() {
    super.onDestroy();

    if (stop.isEnabled()) {
        stop();}
    if (stop2.isEnabled()) {
        stop2();}
    if (stop3.isEnabled()) {
        stop3();}
    if (stop4.isEnabled()) {
        stop4();}
    if (stop5.isEnabled()) {
        stop5();}
    if (stop6.isEnabled()) {
        stop6();}
    if (stop7.isEnabled()) {
        stop7();}
    if (stop8.isEnabled()) {
        stop8();}
    if (stop9.isEnabled()) {
        stop9();}
    if (stop10.isEnabled()) {
        stop10();}
    if (stop11.isEnabled()) {
        stop11();}
    if (stop12.isEnabled()) {
        stop12();}

}

```

```
public void onCompletion(MediaPlayer mp) {
    stop();
    stop2();
    stop3();
    stop4();
    stop5();
    stop6();
    stop7();
    stop8();
    stop9();
    stop10();
    stop11();
    stop12();
}

private void play() {
    mp.start();

    play.setEnabled(false);

    stop.setEnabled(true);
}

private void stop() {
    mp.stop();

    stop.setEnabled(false);

    try {
        mp.prepare();
        mp.seekTo(0);
        play.setEnabled(true);
    }
    catch (Throwable t) {
        goBlooeY(t);
    }
}
```

```
private void play2() {
    media.start();

    play2.setEnabled(false);

    stop2.setEnabled(true);
}

private void stop2() {
    media.stop();

    stop2.setEnabled(false);

    try {
        media.prepare();
        media.seekTo(0);
        play2.setEnabled(true);
    }
    catch (Throwable t) {
        goBlooeey(t);
    }
}

private void play3() {
    media3.start();

    play3.setEnabled(false);

    stop3.setEnabled(true);
}

private void stop3() {
    media3.stop();

    stop3.setEnabled(false);

    try {
        media3.prepare();
        media3.seekTo(0);
```

```
        play3.setEnabled(true);
    }
    catch (Throwable t) {
        goBlooey(t);
    }
}

private void play4() {
    media4.start();

    play4.setEnabled(false);

    stop4.setEnabled(true);
}

private void stop4() {
    media4.stop();

    stop4.setEnabled(false);

    try {
        media4.prepare();
        media4.seekTo(0);
        play4.setEnabled(true);
    }
    catch (Throwable t) {
        goBlooey(t);
    }
}

private void play5() {
    media5.start();

    play5.setEnabled(false);

    stop5.setEnabled(true);
}

private void stop5() {
```

```
media5.stop();

stop5.setEnabled(false);

try {
    media5.prepare();
    media5.seekTo(0);
    play5.setEnabled(true);
}
catch (Throwable t) {
    goBlooney(t);
}

private void play6() {
    media6.start();

    play6.setEnabled(false);

    stop6.setEnabled(true);
}

private void stop6() {
    media6.stop();

    stop6.setEnabled(false);

    try {
        media6.prepare();
        media6.seekTo(0);
        play6.setEnabled(true);
    }
    catch (Throwable t) {
        goBlooney(t);
    }
}

private void play7() {
    media7.start();
```

```
        play7.setEnabled(false);

        stop7.setEnabled(true);
    }

    private void stop7() {
        media7.stop();

        stop7.setEnabled(false);

        try {
            media7.prepare();
            media7.seekTo(0);
            play7.setEnabled(true);
        }
        catch (Throwable t) {
            goBlooey(t);
        }
    }

    private void play8() {
        media8.start();

        play8.setEnabled(false);

        stop8.setEnabled(true);
    }

    private void stop8() {
        media8.stop();

        stop8.setEnabled(false);

        try {
            media8.prepare();
            media8.seekTo(0);
            play8.setEnabled(true);
        }
        catch (Throwable t) {
            goBlooey(t);
        }
    }
}
```

```
    }  
}  
  
private void play9() {  
    media9.start();  
  
    play9.setEnabled(false);  
  
    stop9.setEnabled(true);  
}  
  
private void stop9() {  
    media9.stop();  
  
    stop9.setEnabled(false);  
  
    try {  
        media9.prepare();  
        media9.seekTo(0);  
        play9.setEnabled(true);  
    }  
    catch (Throwable t) {  
        goBlooley(t);  
    }  
}  
  
private void play10() {  
    media10.start();  
  
    play10.setEnabled(false);  
  
    stop10.setEnabled(true);  
}  
  
private void stop10() {  
    media10.stop();  
  
    stop10.setEnabled(false);  
}
```

```
        try {
            media10.prepare();
            media10.seekTo(0);
            play10.setEnabled(true);
        }
        catch (Throwable t) {
            goBlooey(t);
        }
    }

    private void play11() {
        media11.start();

        play11.setEnabled(false);

        stop11.setEnabled(true);
    }

    private void stop11() {
        media11.stop();

        stop11.setEnabled(false);

        try {
            media11.prepare();
            media11.seekTo(0);
            play11.setEnabled(true);
        }
        catch (Throwable t) {
            goBlooey(t);
        }
    }

    private void play12() {
        media12.start();

        play12.setEnabled(false);

        stop12.setEnabled(true);
    }
}
```



```
private void stop12() {
    media12.stop();

    stop12.setEnabled(false);

    try {
        media12.prepare();
        media12.seekTo(0);
        play12.setEnabled(true);
    }
    catch (Throwable t) {
        goBlooney(t);
    }
}

private void loadClip() {
    try {
        mp=MediaPlayer.create(this,
R.raw.papatsiji);
        mp.setOnCompletionListener(this);

        media=MediaPlayer.create(this,
R.raw.pitulas);
        media.setOnCompletionListener(this);

        media3=MediaPlayer.create(this,
R.raw.sepuluh);
        media3.setOnCompletionListener(this);

        media4=MediaPlayer.create(this,
R.raw.selawe);
        media4.setOnCompletionListener(this);

        media5=MediaPlayer.create(this,
R.raw.papatpat);
        media5.setOnCompletionListener(this);
```

```
        media6=MediaPlayer.create(this,
R.raw.papattelu);
media6.setOnCompletionListener(this);

        media7=MediaPlayer.create(this,
R.raw.papatlimo);
media7.setOnCompletionListener(this);

        media8=MediaPlayer.create(this,
R.raw.sebelas);
media8.setOnCompletionListener(this);

        media9=MediaPlayer.create(this,
R.raw.papatenem);
media9.setOnCompletionListener(this);

        media10=MediaPlayer.create(this,
R.raw.papatpitu);
media10.setOnCompletionListener(this);

        media11=MediaPlayer.create(this,
R.raw.papatwolu);
media11.setOnCompletionListener(this);

        media12=MediaPlayer.create(this,
R.raw.empatbelas);
media12.setOnCompletionListener(this);
    }
    catch (Throwable t) {
        goBlooey(t);
    }
}

private void setup() {
```

```

        loadClip();
        play.setEnabled(true);
        stop.setEnabled(false);
        play2.setEnabled(true);
        stop2.setEnabled(false);
        play3.setEnabled(true);
        stop3.setEnabled(false);
        play4.setEnabled(true);
        stop4.setEnabled(false);
        play5.setEnabled(true);
        stop5.setEnabled(false);
        play6.setEnabled(true);
        stop6.setEnabled(false);
        play7.setEnabled(true);
        stop7.setEnabled(false);
        play8.setEnabled(true);
        stop8.setEnabled(false);
        play9.setEnabled(true);
        stop9.setEnabled(false);
        play10.setEnabled(true);
        stop10.setEnabled(false);
        play11.setEnabled(true);
        stop11.setEnabled(false);
        play12.setEnabled(true);
        stop12.setEnabled(false);
    }

    private void goBlooeey(Throwable t) {
        AlertDialog.Builder builder=new
        AlertDialog.Builder(this);

        builder
            .setTitle("Exception!")
            .setMessage(t.toString())
            .setPositiveButton("OK", null)
            .show();
    }
}

```

20.) Potongan kode *class* PsPabean.

```
public class PsPabean extends Activity {

    @Override
    protected void onCreate(Bundle
savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.pspabean);

        WebView browser = (WebView)
findViewById (R.id.pabean);

        browser.getSettings().setJavaScriptEnabled(true);

        browser.loadUrl("file:///android_asset/RutePer
jalanan.html");
    }
}
```

21.) Potongan kode RutePerjalanan.html

```
<!DOCTYPE html>
<html>
  <head>
    <meta name="viewport" content="initial-
scale=1.0, user-scalable=no">
    <meta charset="utf-8">
    <title>Simple markers</title>
    <style>
      html, body,
      #map-canvas {
        height: 90%;
        margin: 0px;
        padding: 0px }

      #btn{
        height: 31px;
        background: #267BA8;
        border: none;
```

```

        padding: 5px;
        color: #fff;
    }
    .input {
        height: 23px;
        padding: 5px;
        width: 150px;
    }

    body {
        text-align:left;
    }

    .wrap {
        background: #f3f8fb;
        border: 3px dashed #61b3de;
        border-radius:4px;

    }
</style>
<script
src="https://maps.googleapis.com/maps/api/js?v
=3.exp&sensor=false"></script>
<script>
    var directionsDisplay;
    var directionsService = new
google.maps.DirectionsService();
    var map;

    function initialize() {
        directionsDisplay = new
google.maps.DirectionsRenderer();
        var myLatLng = new google.maps.LatLng(-
7.235263, 112.741828);
        var mapOptions = {
            zoom: 16,
            center: myLatLng
        }

```

```

    map = new
google.maps.Map(document.getElementById('map-
canvas'), mapOptions);
    directionsDisplay.setMap(map);
    var marker = new google.maps.Marker({
        position: myLatLng,
        map: map,
        title: 'Hello World!'
    });
}

function calcRoute() {
    var start =
document.getElementById('start').value;
    var end =
document.getElementById('end').value;
    var request = {
        origin:start,
        destination:end,
        travelMode:
google.maps.TravelMode.DRIVING
    };
    directionsService.route(request,
function(response, status) {
    if (status ==
google.maps.DirectionsStatus.OK) {

directionsDisplay.setDirections(response);
    }
    });
}

google.maps.event.addDomListener(window,
'load', initialize);

</script>
</head>
<body>
<div class="wrap">
<div id="panel">
<form>

```

```
    Awal:<input class="input" id="start"
type="text" value="">
    <br/>
    Tujuan:<input class="input" id="end"
type="text" disabled="disabled" value="Pasar
Pabean, surabaya">
    <br/>
    <input id="btn" type="button"
value="Search" onclick="calcRoute()">
    </form>
</div>
</div>
<div id="map-canvas"></div>
</body>
</html>
```

Daftar Pustaka

- [1] M.Par. Drs. I Putu Anom, "Analisis Pariwisata," vol. 8 No 2 ISSN 1410 - 3729, p. 1, Juli 2008.
- [2] Renaldy Rakhman Luthfi. (2013) Jurnal Ilmiah Mahasiswa FEB UB - Universitas Brawijaya. [Online].
<http://jimfeb.ub.ac.id/index.php/jimfeb/article>
- [3] Heru Febrianto. (2014, Oktober) SindoNews. Uni Eropa Gelar Pameran Wisata dan Budaya Pertama di RI. [Online].
<http://ekbis.sindonews.com/read/911119/34/uni-eropa-gelar-pameran-wisata-dan-budaya-pertama-di-ri-1413212510>
- [4] SejarahBudaya.[Online].
<http://www.wacananusantara.org/category/sejarah-budaya/seni-tradisi/>
- [5] (2015, Februari) Badan Pusat Statistik. [Online].
<http://www.bps.go.id/Brs/view/id/1121>
- [6] János Csapó, "The Role and Importance of Cultural Tourism," April 2012. [Online]. János Csapó
- [7] Tri Joko Daryanto, Murtanti jani Rahayu1 Istijabatul Aliyah, "Peran Pasar Tradisional Dalam Mendukung Pengembangan Pariwisata Kota Surakarta," *GEMA TEKNIK Majalah Ilmiah Teknik* , 2009.
- [8] Yasser Paragian. (2014, April) TECHINASIA. [Online].
<http://id.techinasia.com/jumlah-pecandu-smartphone-semakin-bertambah-berita-baguskah/>
- [9] Ringga Jati K., "Aplikasi Pemandu Lokasi Objek Wisata dengan Google Maps dan GPS di Kota Solo pada Perangkat Mobile Android," no. 2012, April 2013.
- [10] Venia, and Setia Wirawan Rachmawati, "Aplikasi Web Pemetaan Informasi Wisata Sejarah dan Budaya pada Wilayah DKI Jakarta.," 2012.

- [11] Roby Ardiwidjaja. Academia.edu. [Online].
https://www.academia.edu/4929428/PARIWISATA_BUDAYA_sebagai_salah_satu_alat_pelestari_kesenian_tradisional
- [12] (2012, April) Lensa Indonesia. [Online].
www.lensaindonesia.com
- [13] Pasar Surya Surabaya. (2013, Januari) Pasar Surya Service to Care. [Online].
<http://pasarsurya.com/index.php/wisata-belanja/ikan/8-pasar-pabean-pasar-terbesar-di-asia-tenggara>
- [14] Wisata Surabaya Kota Metropolitan Dengan Sejuta Budaya. [Online].
<http://www.eastjava.com/tourism/surabaya/ina/>
- [15] Website Resmi Surabaya. [Online].
<http://www.surabaya.go.id/profilkota/index.php?id=24>
- [16] Aingindra. (2013, Desember) Media Informasi Online Seputar Ternologi Terkini. [Online].
<http://www.aingindra.com/android-adalah-pengertian-android-sistem-operasi.html>
- [17] [Online]. <http://developer.android.com/index.html>
- [18] Developer Android. [Online].
<http://developer.android.com/index.html>
- [19] Waterfall Model ADVANTAGES, EXAMPLES, PHASES AND MORE ABOUT SOFTWARE DEVELOPMENT . [Online].
<http://www.waterfall-model.com/>
- [20] Melonfire. (2006, September) TechRepublic.com. [Online]. 22
- [21] (2015) <http://www.antutu.com/index.shtml>.
- [22] DynamicApps. (2015) Google Play. [Online].
<https://play.google.com/store/apps/details?id=com.inter.net.speed.meter.lite>

BIODATA PENULIS



Penulis dilahirkan di Surabaya pada tanggal 14 Januari 1993. Penulis merupakan anak kedua dari dua bersaudara. Penulis menempuh pendidikan di SDN Kebraon III Surabaya, SMP Negeri 16 Surabaya dan SMK Negeri 1 Surabaya. Pada tahun 2011 penulis diterima di jurusan Sistem Informasi – Institut Teknologi Sepuluh Nopember (ITS) dan terdaftar dengan NRP 5211100103.

Selain kesibukan akademik, penulis juga pernah mengikuti berbagai kegiatan organisasi diluar kampus. Penulis pernah terlibat dalam proyek SKPD Monitoring dan Evaluasi Informasi Teknologi Kecamatan dan Dinas Surabaya 2014-2015 . Pernah mengikuti pelatihan Uji Kompetensi Keahlian meliputi DNS Server, Web Server, Dan Instalasi OS(Sistem Operasi : Linux Debian).

(halaman ini sengaja dikosongkan)